FIG. 1

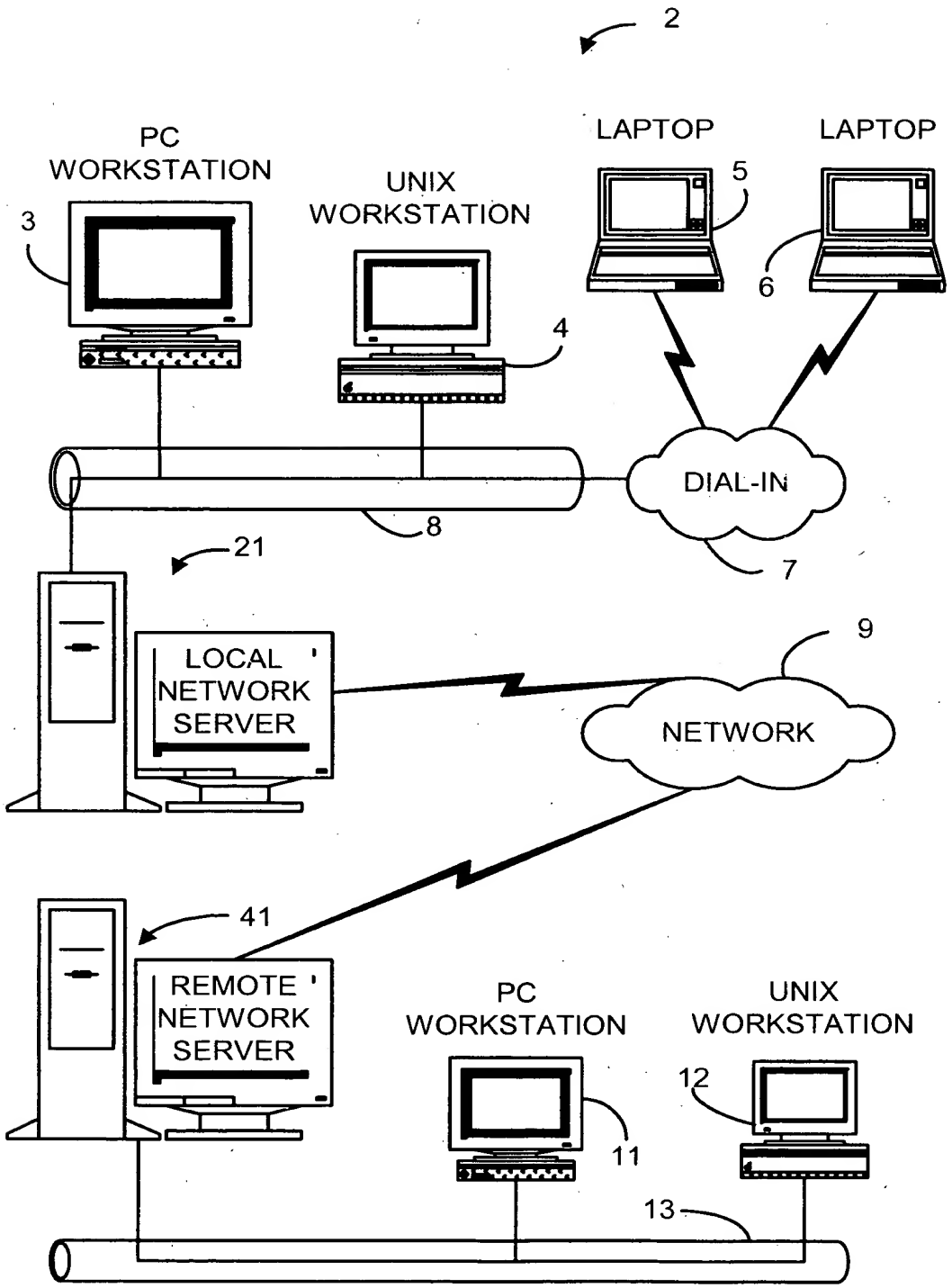

2
PC WORKSTATION
3
UNIX WORKSTATION
4
LAPTOP
5
LAPTOP
6
DIAL-IN
7
8
21
LOCAL NETWORK SERVER
9
NETWORK
41
REMOTE NETWORK SERVER
PC WORKSTATION
11
UNIX WORKSTATION
12
13

FIG. 2

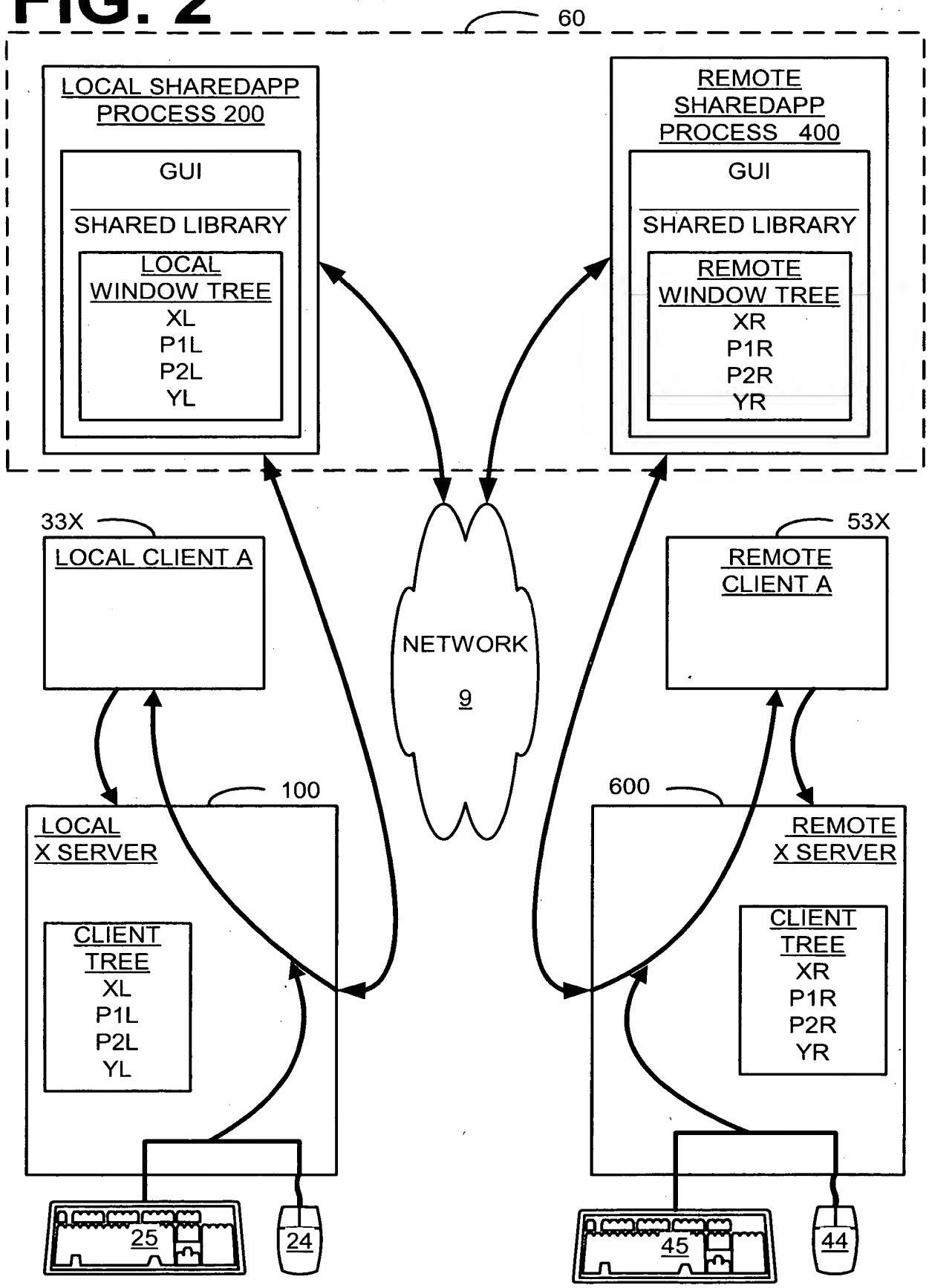

60
LOCAL SHAREDAPP PROCESS 200
GUI
SHARED LIBRARY
LOCAL WINDOW TREE
XL
P1L
P2L
YL
REMOTE SHAREDAPP PROCESS 400
GUI
SHARED LIBRARY
REMOTE WINDOW TREE
XR
P1R
P2R
YR
33X
LOCAL CLIENT A
NETWORK
9
53X
REMOTE CLIENT A
100
LOCAL X SERVER
CLIENT TREE
XL
P1L
P2L
YL
600
REMOTE X SERVER
CLIENT TREE
XR
P1R
P2R
YR
25
24
45
44

FIG. 3A

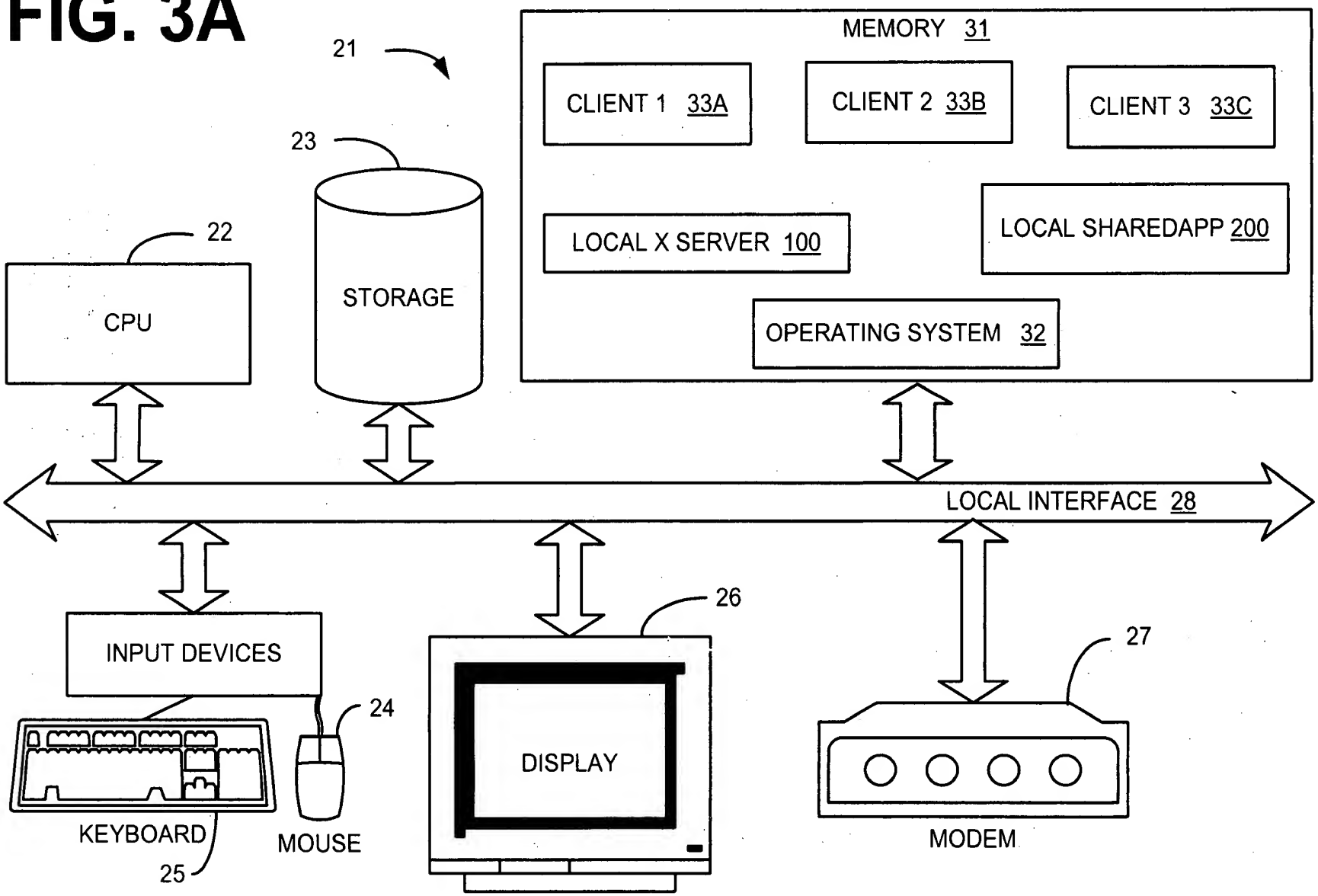

21
MEMORY 31
CLIENT 1 33A
CLIENT 2 33B
CLIENT 3 33C
LOCAL X SERVER 100
LOCAL SHAREDAPP 200
OPERATING SYSTEM 32
23
22
CPU
STORAGE
LOCAL INTERFACE 28
INPUT DEVICES
26
27
24
DISPLAY
KEYBOARD
MOUSE
25
MODEM

FIG. 3B

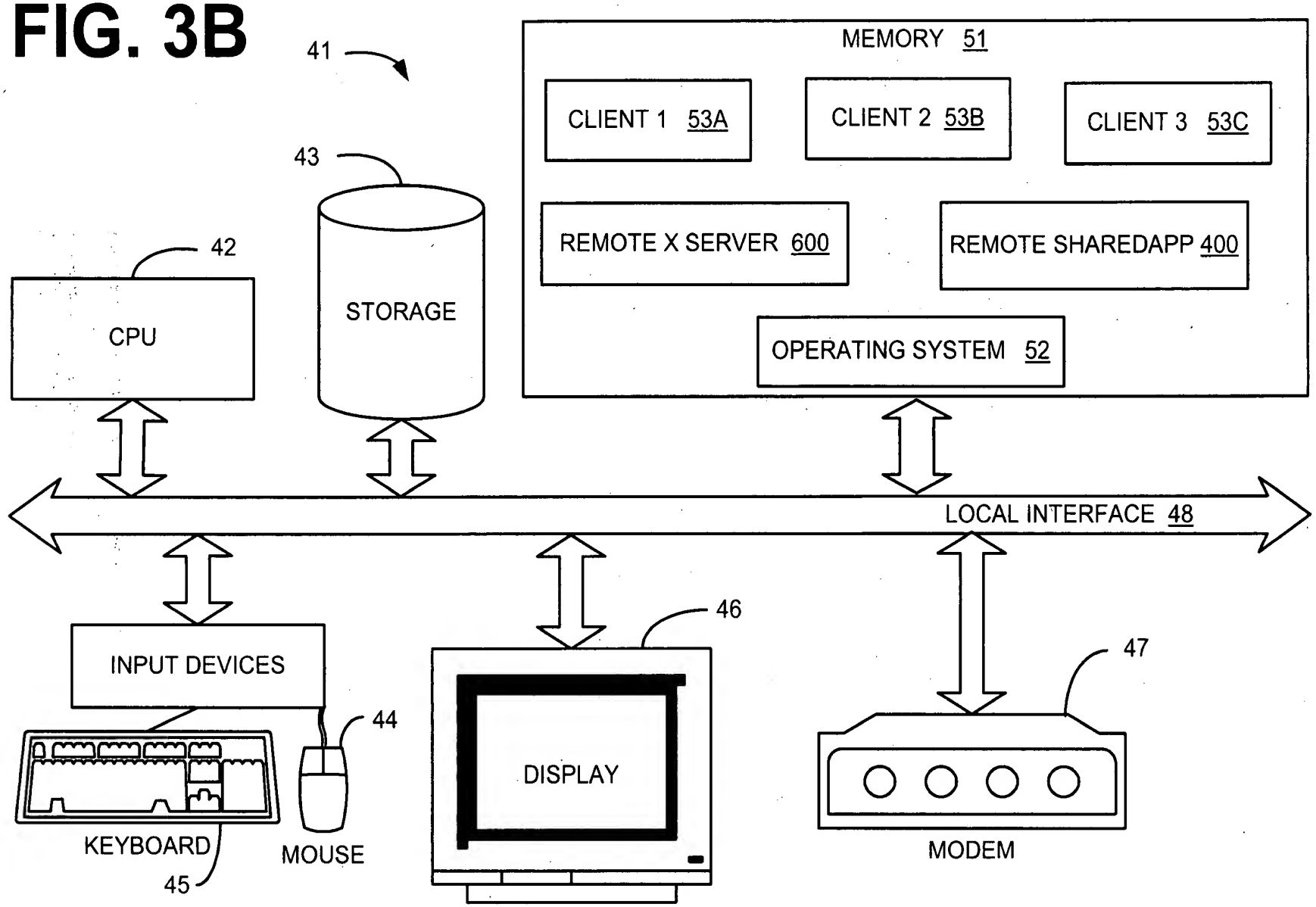

41
MEMORY 51
CLIENT 1 53A
CLIENT 2 53B
CLIENT 3 53C
REMOTE X SERVER 600
REMOTE SHAREDAPP 400
OPERATING SYSTEM 52
43
STORAGE
42
CPU
LOCAL INTERFACE 48
INPUT DEVICES
44
KEYBOARD
45
MOUSE
46
DISPLAY
47
MODEM

FIG. 4

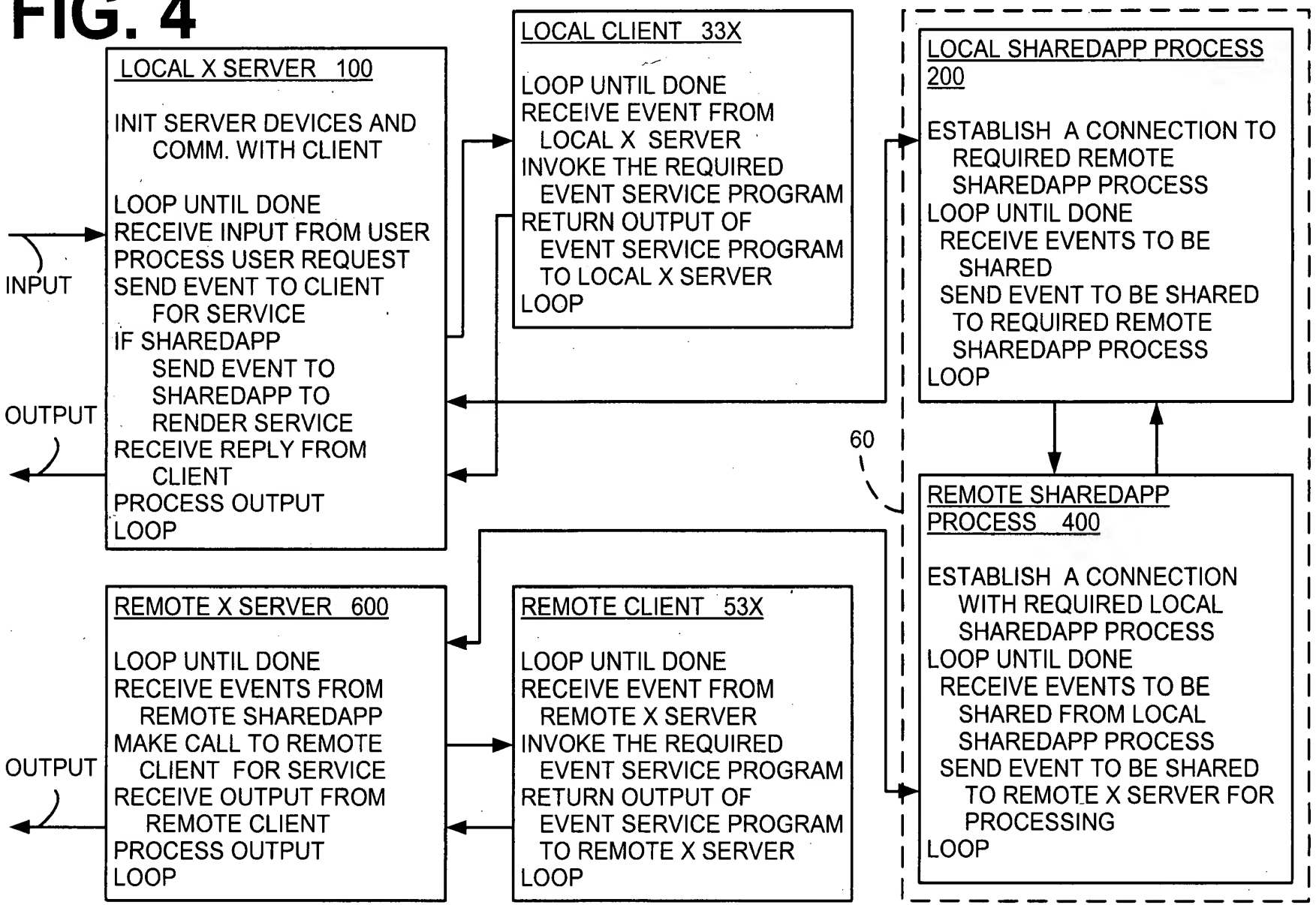

LOCAL X SERVER 100
INIT SERVER DEVICES AND COMM. WITH CLIENT
LOOP UNTIL DONE
RECEIVE INPUT FROM USER
PROCESS USER REQUEST
SEND EVENT TO CLIENT FOR SERVICE
IF SHAREDAPP SEND EVENT TO SHAREDAPP TO RENDER SERVICE
RECEIVE REPLY FROM CLIENT
PROCESS OUTPUT
LOOP
INPUT
OUTPUT
LOCAL CLIENT 33X
LOOP UNTIL DONE
RECEIVE EVENT FROM LOCAL X SERVER
INVOKE THE REQUIRED EVENT SERVICE PROGRAM
RETURN OUTPUT OF EVENT SERVICE PROGRAM TO LOCAL X SERVER
LOOP
LOCAL SHAREDAPP PROCESS 200
ESTABLISH A CONNECTION TO REQUIRED REMOTE SHAREDAPP PROCESS
LOOP UNTIL DONE
RECEIVE EVENTS TO BE SHARED
SEND EVENT TO BE SHARED TO REQUIRED REMOTE SHAREDAPP PROCESS
LOOP
60
REMOTE SHAREDAPP PROCESS 400
ESTABLISH A CONNECTION WITH REQUIRED LOCAL SHAREDAPP PROCESS
LOOP UNTIL DONE
RECEIVE EVENTS TO BE SHARED FROM LOCAL SHAREDAPP PROCESS
SEND EVENT TO BE SHARED TO REMOTE X SERVER FOR PROCESSING
LOOP
REMOTE X SERVER 600
LOOP UNTIL DONE
RECEIVE EVENTS FROM REMOTE SHAREDAPP
MAKE CALL TO REMOTE CLIENT FOR SERVICE
RECEIVE OUTPUT FROM REMOTE CLIENT
PROCESS OUTPUT
LOOP
OUTPUT
REMOTE CLIENT 53X
LOOP UNTIL DONE
RECEIVE EVENT FROM REMOTE X SERVER
INVOKE THE REQUIRED EVENT SERVICE PROGRAM
RETURN OUTPUT OF EVENT SERVICE PROGRAM TO REMOTE X SERVER
LOOP

FIG. 5A

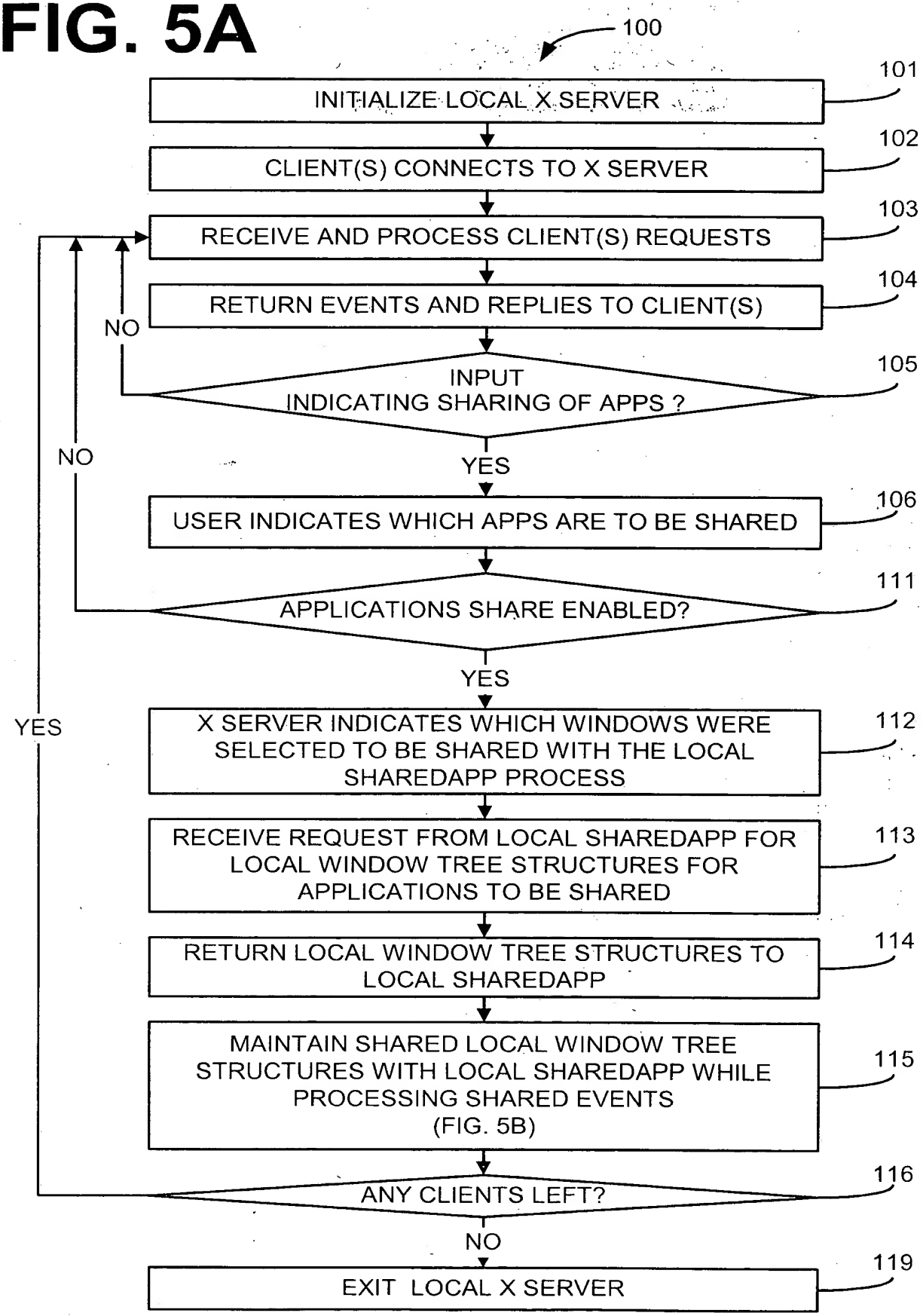

100
INITIALIZE LOCAL X SERVER
101
CLIENT(S) CONNECTS TO X SERVER
102
RECEIVE AND PROCESS CLIENT(S) REQUESTS
103
RETURN EVENTS AND REPLIES TO CLIENT(S)
104
NO
INPUT INDICATING SHARING OF APPS ?
105
NO
YES
USER INDICATES WHICH APPS ARE TO BE SHARED
106
APPLICATIONS SHARE ENABLED?
111
YES
YES
X SERVER INDICATES WHICH WINDOWS WERE SELECTED TO BE SHARED WITH THE LOCAL SHAREDAPP PROCESS
112
RECEIVE REQUEST FROM LOCAL SHAREDAPP FOR LOCAL WINDOW TREE STRUCTURES FOR APPLICATIONS TO BE SHARED
113
RETURN LOCAL WINDOW TREE STRUCTURES TO LOCAL SHAREDAPP
114
MAINTAIN SHARED LOCAL WINDOW TREE STRUCTURES WITH LOCAL SHAREDAPP WHILE PROCESSING SHARED EVENTS (FIG. 5B)
115
ANY CLIENTS LEFT?
116
NO
EXIT LOCAL X SERVER
119

FIG. 5B

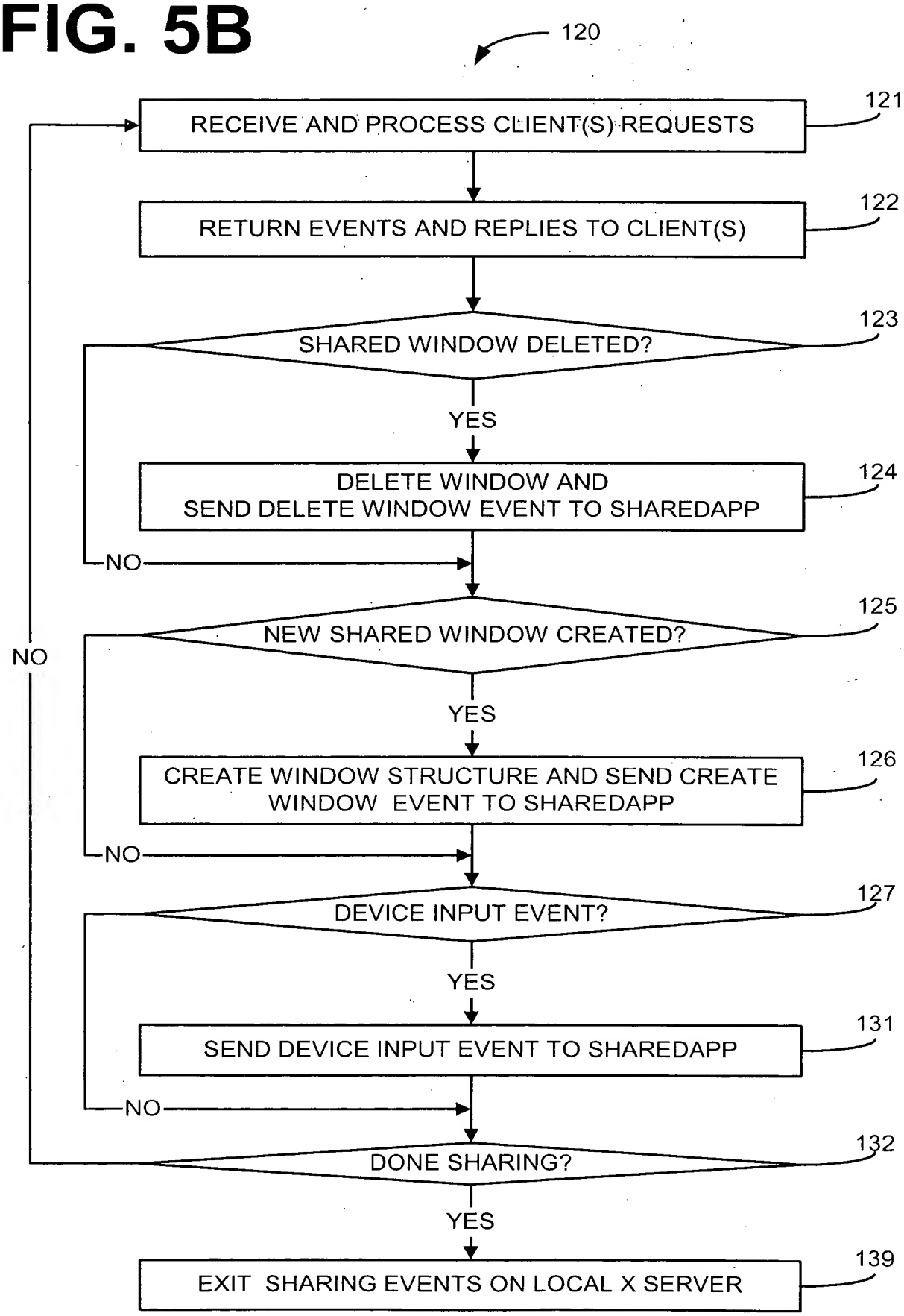

120
RECEIVE AND PROCESS CLIENT(S) REQUESTS
121
RETURN EVENTS AND REPLIES TO CLIENT(S)
122
SHARED WINDOW DELETED?
123
YES
DELETE WINDOW AND
SEND DELETE WINDOW EVENT TO SHAREDAPP
124
NO
NEW SHARED WINDOW CREATED?
125
NO
YES
CREATE WINDOW STRUCTURE AND SEND CREATE
WINDOW EVENT TO SHAREDAPP
126
NO
DEVICE INPUT EVENT?
127
YES
SEND DEVICE INPUT EVENT TO SHAREDAPP
131
NO
DONE SHARING?
132
YES
EXIT SHARING EVENTS ON LOCAL X SERVER
139

FIG. 6A

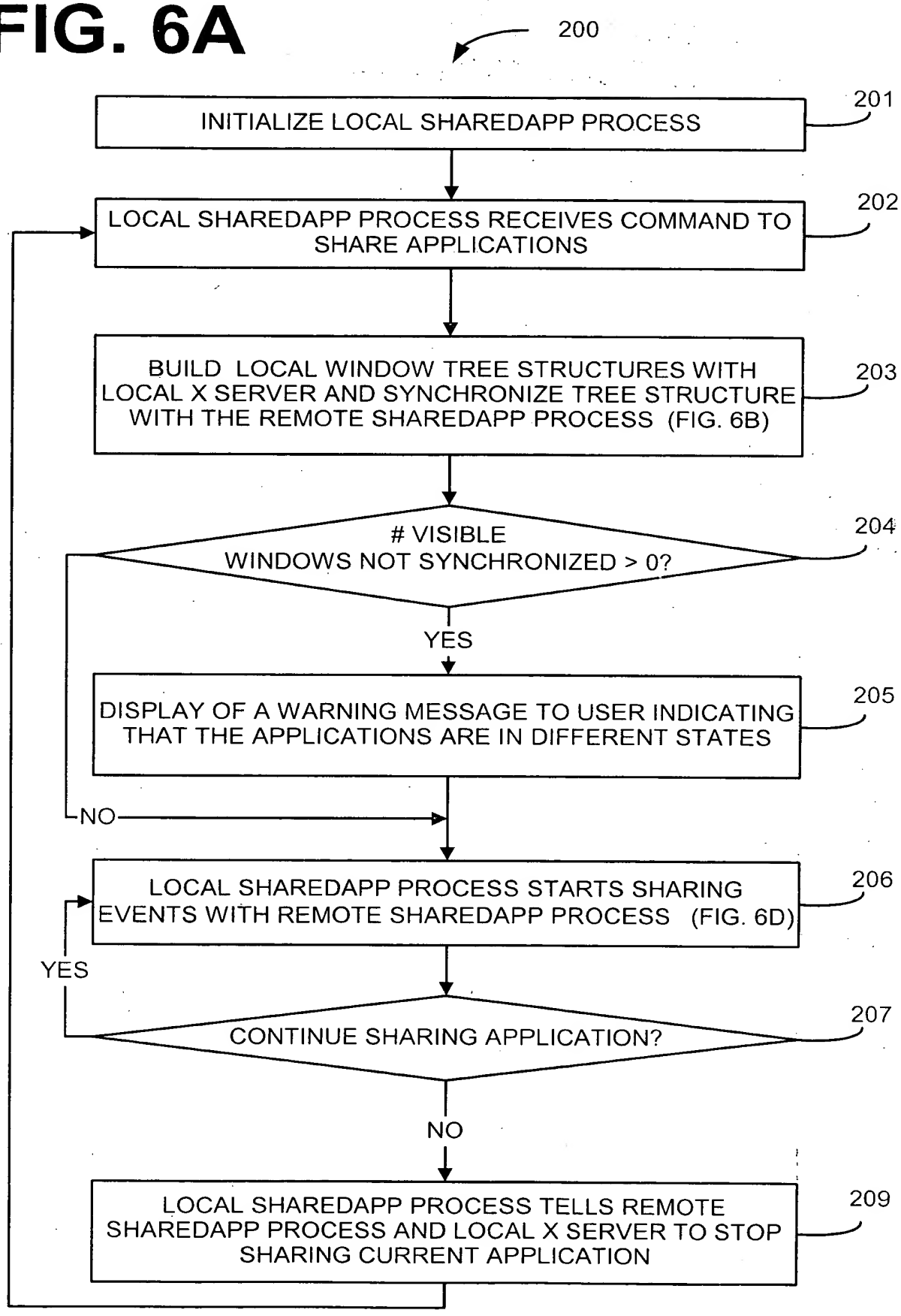

200
INITIALIZE LOCAL SHAREDAPP PROCESS
201
LOCAL SHAREDAPP PROCESS RECEIVES COMMAND TO SHARE APPLICATIONS
202
BUILD LOCAL WINDOW TREE STRUCTURES WITH LOCAL X SERVER AND SYNCHRONIZE TREE STRUCTURE WITH THE REMOTE SHAREDAPP PROCESS (FIG. 6B)
203
VISIBLE WINDOWS NOT SYNCHRONIZED > 0?
204
YES
DISPLAY OF A WARNING MESSAGE TO USER INDICATING THAT THE APPLICATIONS ARE IN DIFFERENT STATES
205
NO
LOCAL SHAREDAPP PROCESS STARTS SHARING EVENTS WITH REMOTE SHAREDAPP PROCESS (FIG. 6D)
206
YES
CONTINUE SHARING APPLICATION?
207
NO
LOCAL SHAREDAPP PROCESS TELLS REMOTE SHAREDAPP PROCESS AND LOCAL X SERVER TO STOP SHARING CURRENT APPLICATION
209

FIG. 6B

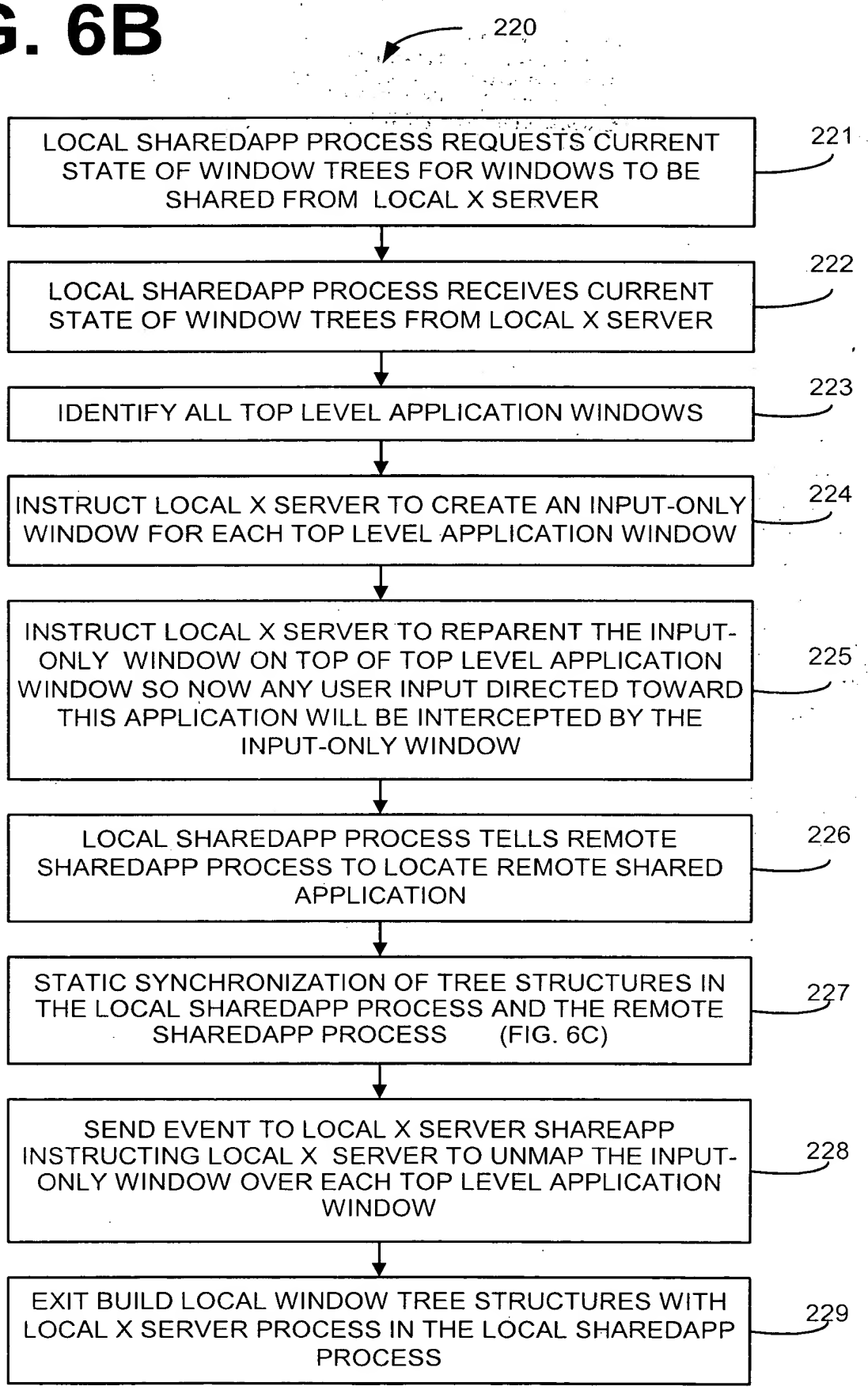

220
LOCAL SHAREDAPP PROCESS REQUESTS CURRENT STATE OF WINDOW TREES FOR WINDOWS TO BE SHARED FROM LOCAL X SERVER
221
LOCAL SHAREDAPP PROCESS RECEIVES CURRENT STATE OF WINDOW TREES FROM LOCAL X SERVER
222
IDENTIFY ALL TOP LEVEL APPLICATION WINDOWS
223
INSTRUCT LOCAL X SERVER TO CREATE AN INPUT-ONLY WINDOW FOR EACH TOP LEVEL APPLICATION WINDOW
224
INSTRUCT LOCAL X SERVER TO REPARENT THE INPUT-ONLY WINDOW ON TOP OF TOP LEVEL APPLICATION WINDOW SO NOW ANY USER INPUT DIRECTED TOWARD THIS APPLICATION WILL BE INTERCEPTED BY THE INPUT-ONLY WINDOW
225
LOCAL SHAREDAPP PROCESS TELLS REMOTE SHAREDAPP PROCESS TO LOCATE REMOTE SHARED APPLICATION
226
STATIC SYNCHRONIZATION OF TREE STRUCTURES IN THE LOCAL SHAREDAPP PROCESS AND THE REMOTE SHAREDAPP PROCESS (FIG. 6C)
227
SEND EVENT TO LOCAL X SERVER SHAREAPP INSTRUCTING LOCAL X SERVER TO UNMAP THE INPUT-ONLY WINDOW OVER EACH TOP LEVEL APPLICATION WINDOW
228
EXIT BUILD LOCAL WINDOW TREE STRUCTURES WITH LOCAL X SERVER PROCESS IN THE LOCAL SHAREDAPP PROCESS
229

FIG. 6C

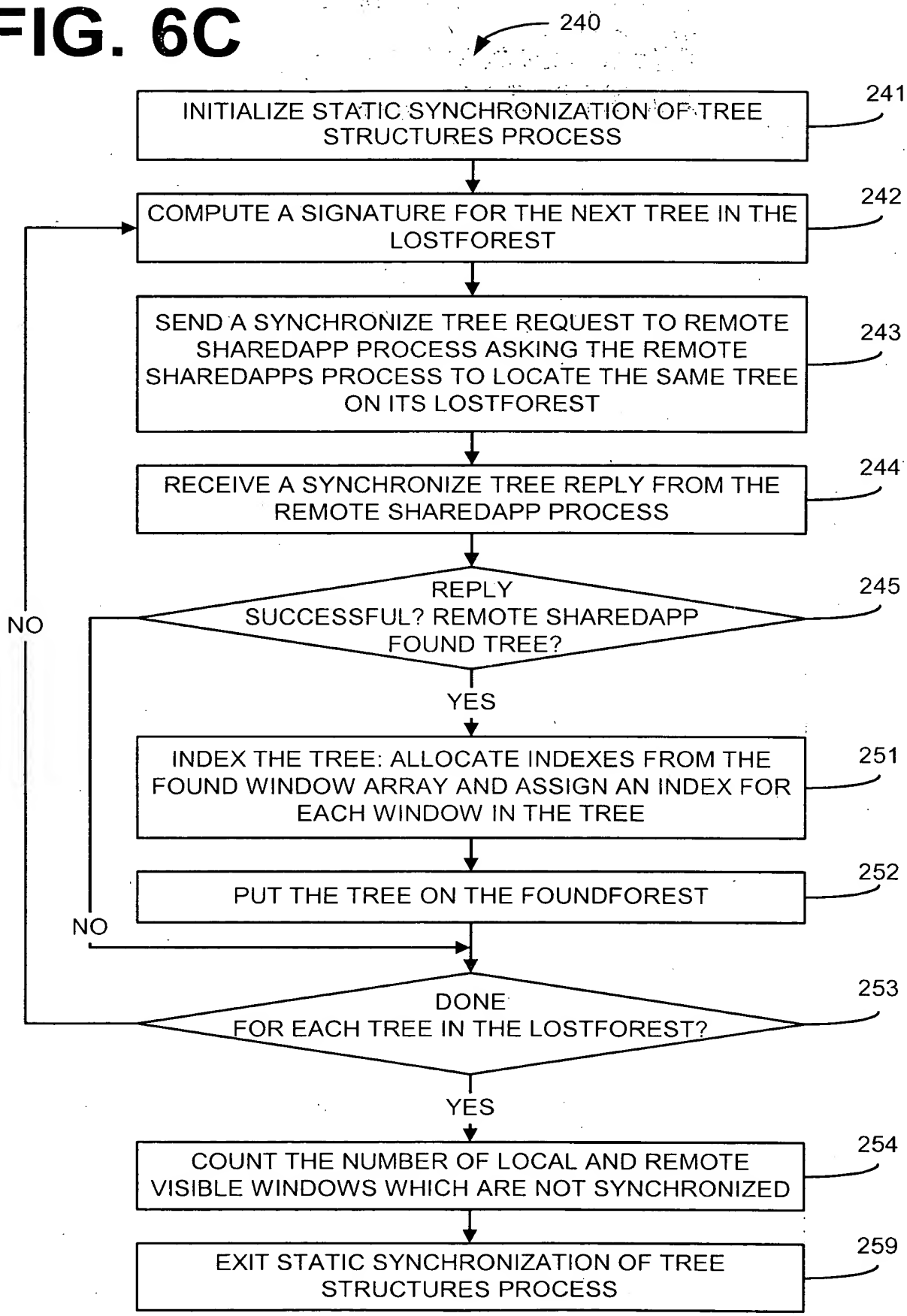

240
INITIALIZE STATIC SYNCHRONIZATION OF TREE STRUCTURES PROCESS
241
COMPUTE A SIGNATURE FOR THE NEXT TREE IN THE LOSTFOREST
242
SEND A SYNCHRONIZE TREE REQUEST TO REMOTE SHAREDAPP PROCESS ASKING THE REMOTE SHAREDAPPS PROCESS TO LOCATE THE SAME TREE ON ITS LOSTFOREST
243
RECEIVE A SYNCHRONIZE TREE REPLY FROM THE REMOTE SHAREDAPP PROCESS
244
REPLY SUCCESSFUL? REMOTE SHAREDAPP FOUND TREE?
245
NO
YES
INDEX THE TREE: ALLOCATE INDEXES FROM THE FOUND WINDOW ARRAY AND ASSIGN AN INDEX FOR EACH WINDOW IN THE TREE
251
PUT THE TREE ON THE FOUNDFOREST
252
NO
DONE FOR EACH TREE IN THE LOSTFOREST?
253
YES
COUNT THE NUMBER OF LOCAL AND REMOTE VISIBLE WINDOWS WHICH ARE NOT SYNCHRONIZED
254
EXIT STATIC SYNCHRONIZATION OF TREE STRUCTURES PROCESS
259

FIG. 6D

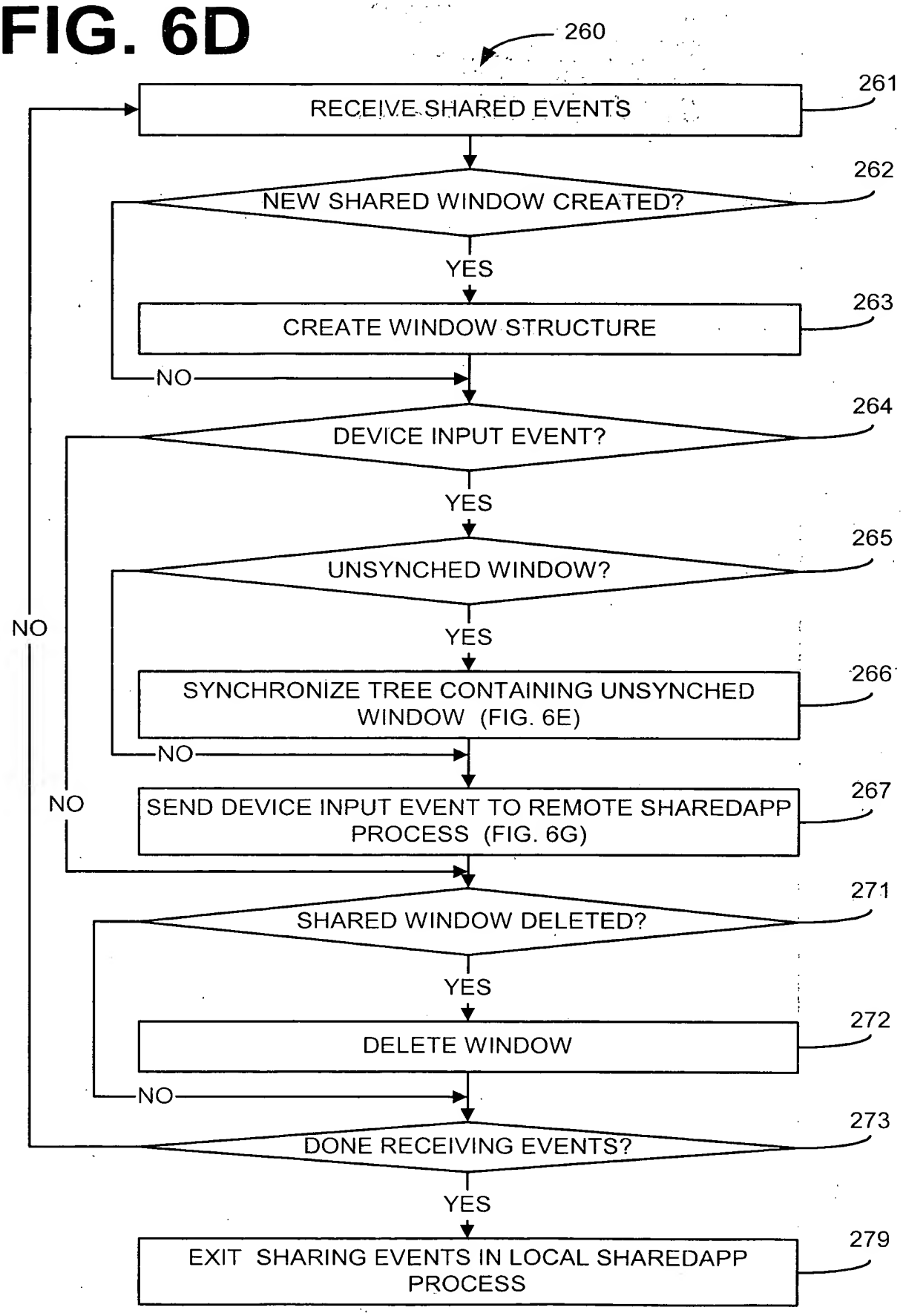

260
RECEIVE SHARED EVENTS
261
NEW SHARED WINDOW CREATED?
262
YES
CREATE WINDOW STRUCTURE
263
NO
DEVICE INPUT EVENT?
264
YES
UNSYNCHED WINDOW?
265
YES
SYNCHRONIZE TREE CONTAINING UNSYNCHED WINDOW (FIG. 6E)
266
NO
SEND DEVICE INPUT EVENT TO REMOTE SHAREDAPP PROCESS (FIG. 6G)
267
NO
NO
SHARED WINDOW DELETED?
271
YES
DELETE WINDOW
272
NO
DONE RECEIVING EVENTS?
273
YES
EXIT SHARING EVENTS IN LOCAL SHAREDAPP PROCESS
279

FIG. 6E

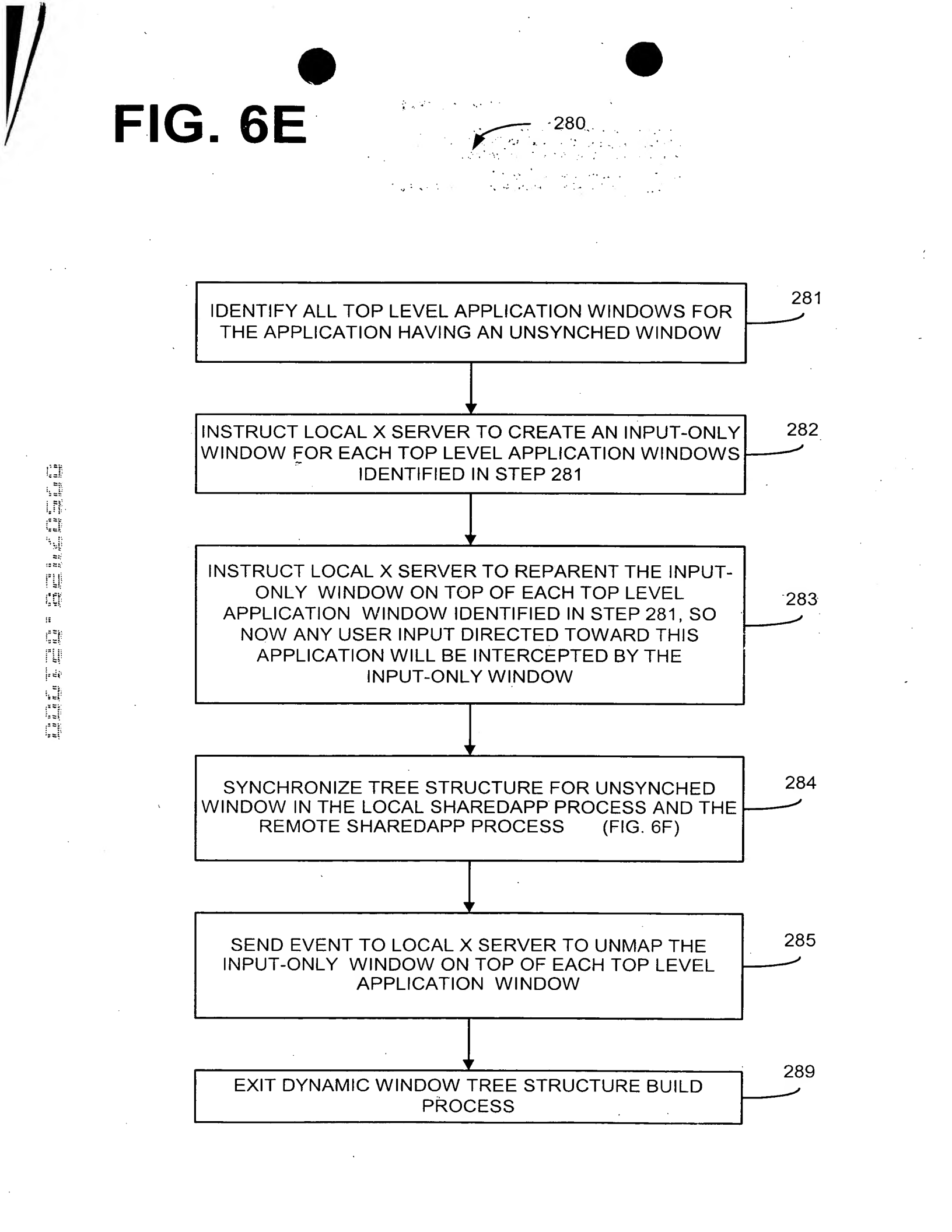

280
IDENTIFY ALL TOP LEVEL APPLICATION WINDOWS FOR THE APPLICATION HAVING AN UNSYNCHED WINDOW
281
INSTRUCT LOCAL X SERVER TO CREATE AN INPUT-ONLY WINDOW FOR EACH TOP LEVEL APPLICATION WINDOWS IDENTIFIED IN STEP 281
282
INSTRUCT LOCAL X SERVER TO REPARENT THE INPUT-ONLY WINDOW ON TOP OF EACH TOP LEVEL APPLICATION WINDOW IDENTIFIED IN STEP 281, SO NOW ANY USER INPUT DIRECTED TOWARD THIS APPLICATION WILL BE INTERCEPTED BY THE INPUT-ONLY WINDOW
283
SYNCHRONIZE TREE STRUCTURE FOR UNSYNCHED WINDOW IN THE LOCAL SHAREDAPP PROCESS AND THE REMOTE SHAREDAPP PROCESS (FIG. 6F)
284
SEND EVENT TO LOCAL X SERVER TO UNMAP THE INPUT-ONLY WINDOW ON TOP OF EACH TOP LEVEL APPLICATION WINDOW
285
EXIT DYNAMIC WINDOW TREE STRUCTURE BUILD PROCESS
289

FIG. 6F

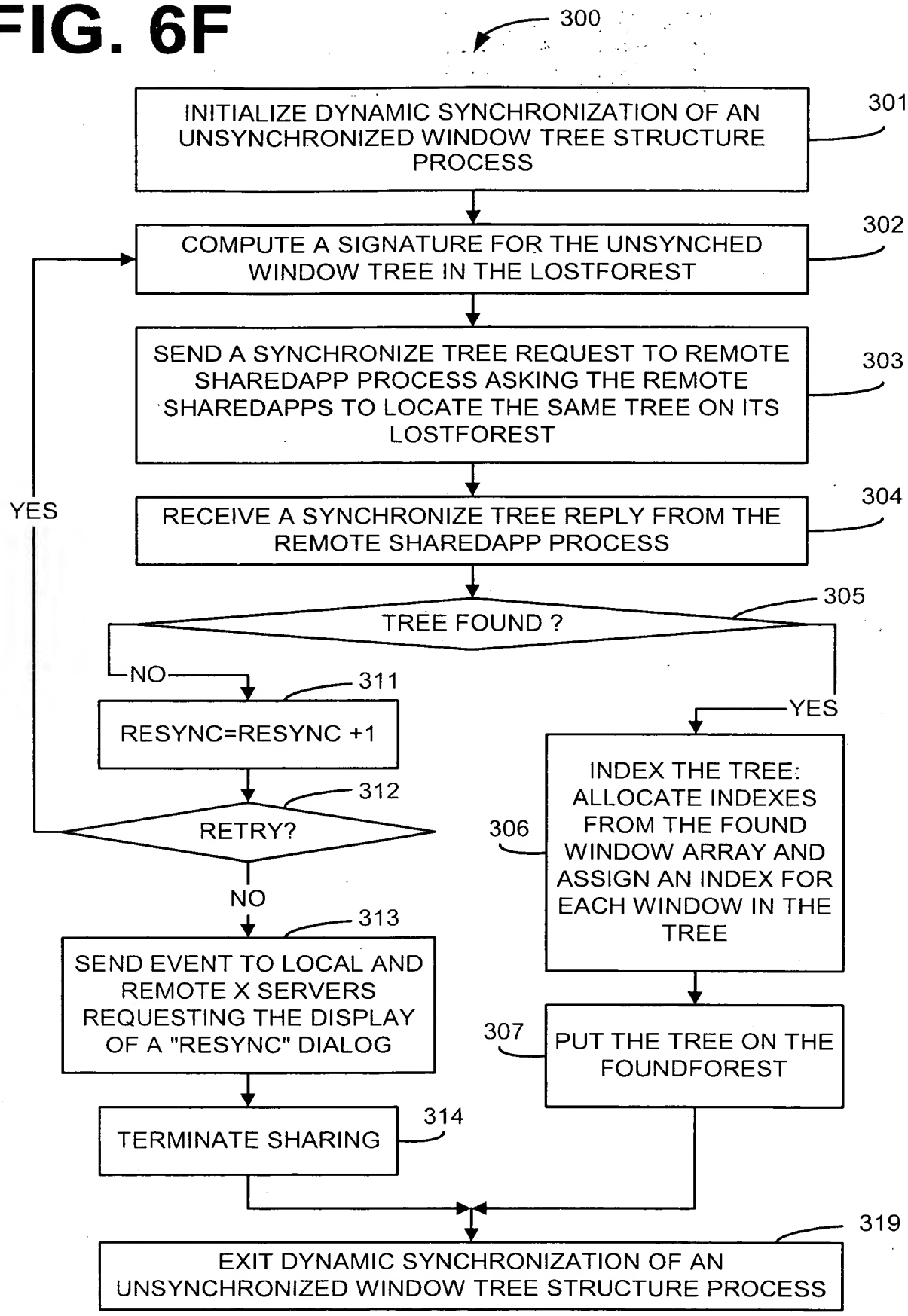

300
INITIALIZE DYNAMIC SYNCHRONIZATION OF AN UNSYNCHRONIZED WINDOW TREE STRUCTURE PROCESS
301
COMPUTE A SIGNATURE FOR THE UNSYNCHED WINDOW TREE IN THE LOSTFOREST
302
SEND A SYNCHRONIZE TREE REQUEST TO REMOTE SHAREDAPP PROCESS ASKING THE REMOTE SHAREDAPPS TO LOCATE THE SAME TREE ON ITS LOSTFOREST
303
RECEIVE A SYNCHRONIZE TREE REPLY FROM THE REMOTE SHAREDAPP PROCESS
304
TREE FOUND ?
305
NO
YES
YES
311
RESYNC=RESYNC +1
312
RETRY?
NO
313
SEND EVENT TO LOCAL AND REMOTE X SERVERS REQUESTING THE DISPLAY OF A "RESYNC" DIALOG
314
TERMINATE SHARING
306
INDEX THE TREE: ALLOCATE INDEXES FROM THE FOUND WINDOW ARRAY AND ASSIGN AN INDEX FOR EACH WINDOW IN THE TREE
307
PUT THE TREE ON THE FOUNDFOREST
319
EXIT DYNAMIC SYNCHRONIZATION OF AN UNSYNCHRONIZED WINDOW TREE STRUCTURE PROCESS

FIG. 6G

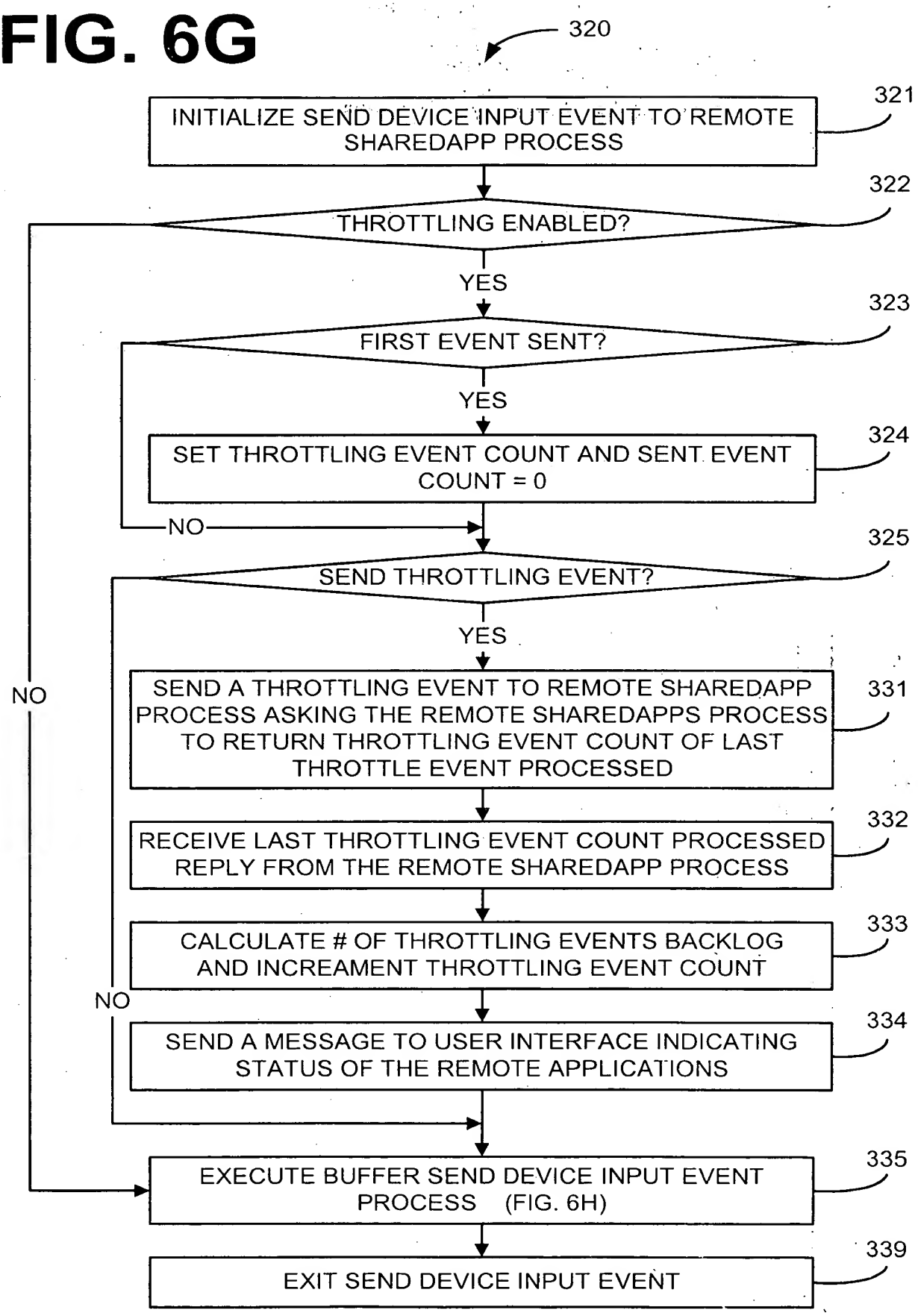

320
INITIALIZE SEND DEVICE INPUT EVENT TO REMOTE SHAREDAPP PROCESS
321
THROTTLING ENABLED?
322
YES
FIRST EVENT SENT?
323
YES
SET THROTTLING EVENT COUNT AND SENT EVENT COUNT = 0
324
NO
SEND THROTTLING EVENT?
325
YES
NO
SEND A THROTTLING EVENT TO REMOTE SHAREDAPP PROCESS ASKING THE REMOTE SHAREDAPPS PROCESS TO RETURN THROTTLING EVENT COUNT OF LAST THROTTLE EVENT PROCESSED
331
RECEIVE LAST THROTTLING EVENT COUNT PROCESSED REPLY FROM THE REMOTE SHAREDAPP PROCESS
332
CALCULATE # OF THROTTLING EVENTS BACKLOG AND INCREAMENT THROTTLING EVENT COUNT
333
NO
SEND A MESSAGE TO USER INTERFACE INDICATING STATUS OF THE REMOTE APPLICATIONS
334
EXECUTE BUFFER SEND DEVICE INPUT EVENT PROCESS (FIG. 6H)
335
EXIT SEND DEVICE INPUT EVENT
339

FIG. 6H

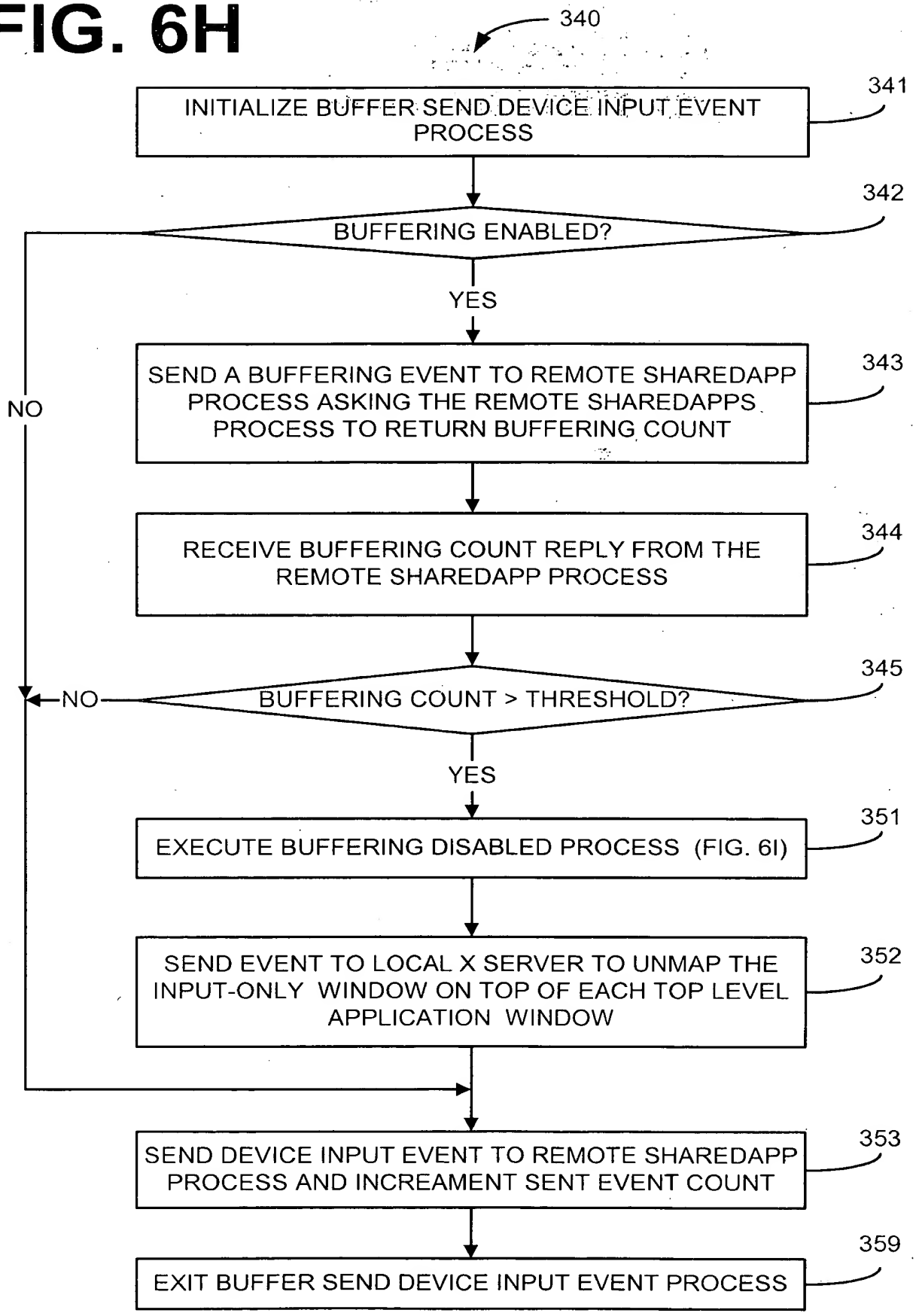

340
INITIALIZE BUFFER SEND DEVICE INPUT EVENT PROCESS
341
BUFFERING ENABLED?
342
YES
NO
SEND A BUFFERING EVENT TO REMOTE SHAREDAPP PROCESS ASKING THE REMOTE SHAREDAPPS PROCESS TO RETURN BUFFERING COUNT
343
RECEIVE BUFFERING COUNT REPLY FROM THE REMOTE SHAREDAPP PROCESS
344
BUFFERING COUNT > THRESHOLD?
345
NO
YES
EXECUTE BUFFERING DISABLED PROCESS (FIG. 6I)
351
SEND EVENT TO LOCAL X SERVER TO UNMAP THE INPUT-ONLY WINDOW ON TOP OF EACH TOP LEVEL APPLICATION WINDOW
352
SEND DEVICE INPUT EVENT TO REMOTE SHAREDAPP PROCESS AND INCREAMENT SENT EVENT COUNT
353
EXIT BUFFER SEND DEVICE INPUT EVENT PROCESS
359

FIG. 6I

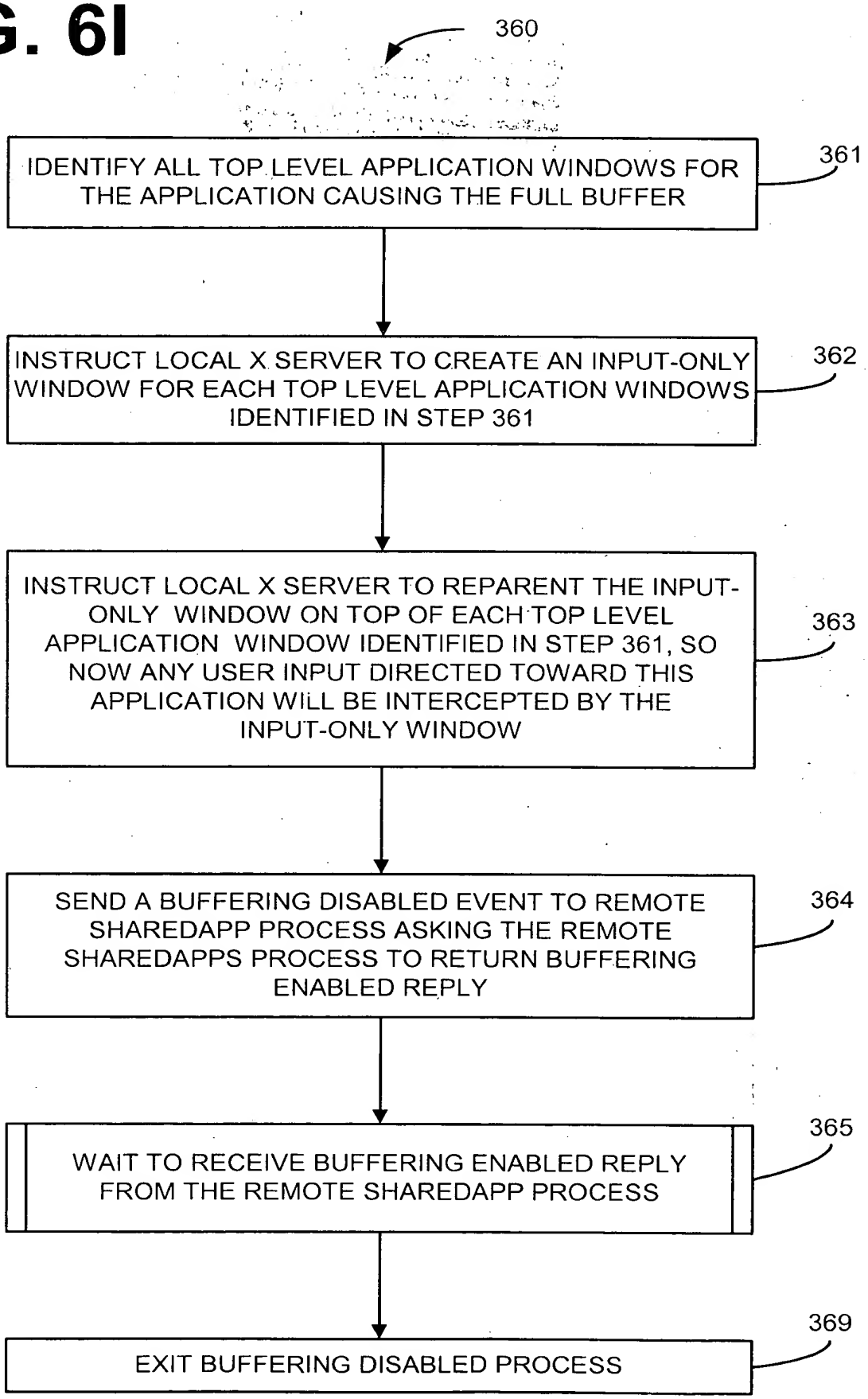

360
IDENTIFY ALL TOP LEVEL APPLICATION WINDOWS FOR THE APPLICATION CAUSING THE FULL BUFFER
361
INSTRUCT LOCAL X SERVER TO CREATE AN INPUT-ONLY WINDOW FOR EACH TOP LEVEL APPLICATION WINDOWS IDENTIFIED IN STEP 361
362
INSTRUCT LOCAL X SERVER TO REPARENT THE INPUT-ONLY WINDOW ON TOP OF EACH TOP LEVEL APPLICATION WINDOW IDENTIFIED IN STEP 361, SO NOW ANY USER INPUT DIRECTED TOWARD THIS APPLICATION WILL BE INTERCEPTED BY THE INPUT-ONLY WINDOW
363
SEND A BUFFERING DISABLED EVENT TO REMOTE SHAREDAPP PROCESS ASKING THE REMOTE SHAREDAPPS PROCESS TO RETURN BUFFERING ENABLED REPLY
364
WAIT TO RECEIVE BUFFERING ENABLED REPLY FROM THE REMOTE SHAREDAPP PROCESS
365
EXIT BUFFERING DISABLED PROCESS
369

FIG. 7A

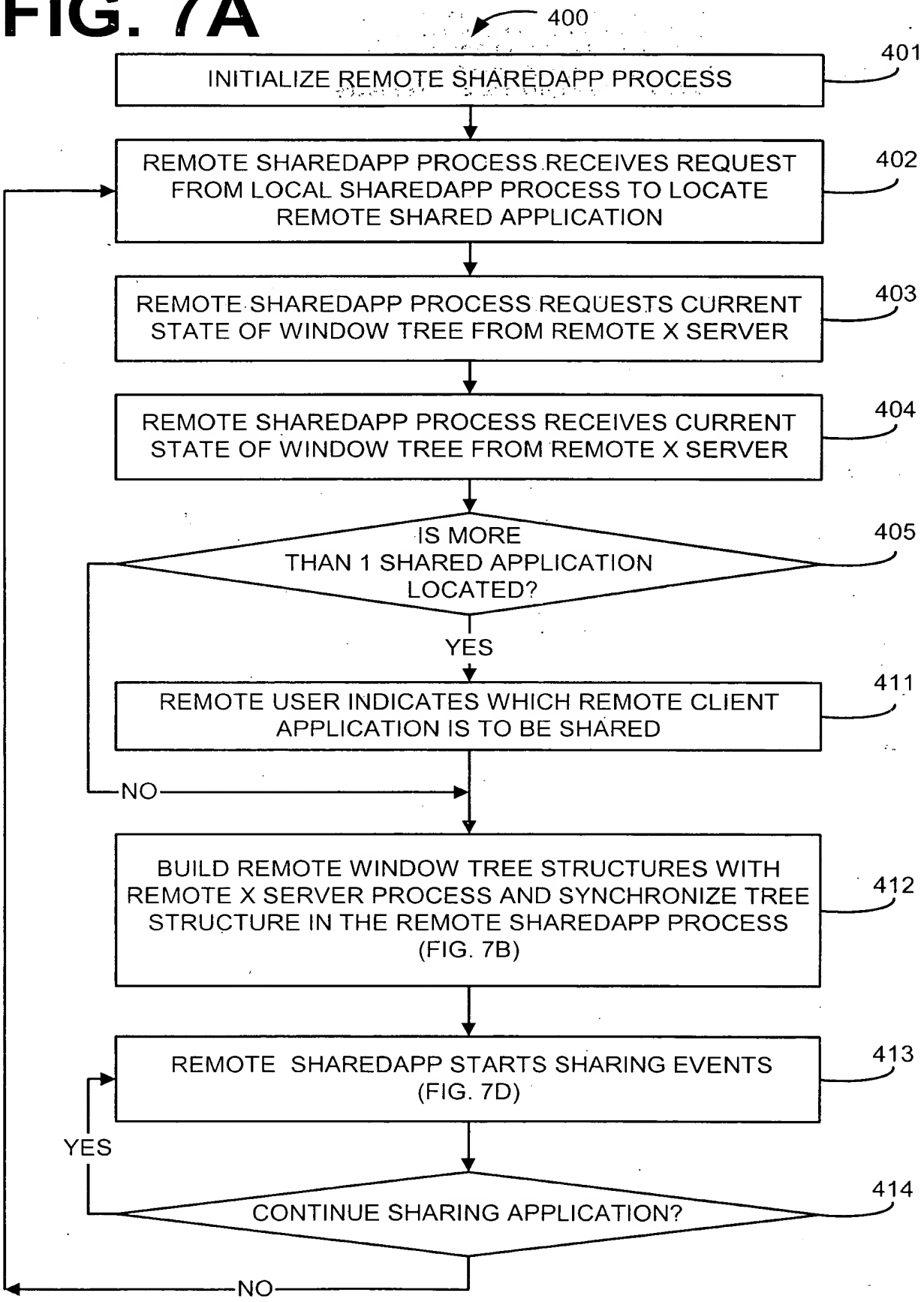

400
INITIALIZE REMOTE SHAREDAPP PROCESS
401
REMOTE SHAREDAPP PROCESS RECEIVES REQUEST FROM LOCAL SHAREDAPP PROCESS TO LOCATE REMOTE SHARED APPLICATION
402
REMOTE SHAREDAPP PROCESS REQUESTS CURRENT STATE OF WINDOW TREE FROM REMOTE X SERVER
403
REMOTE SHAREDAPP PROCESS RECEIVES CURRENT STATE OF WINDOW TREE FROM REMOTE X SERVER
404
IS MORE THAN 1 SHARED APPLICATION LOCATED?
405
YES
REMOTE USER INDICATES WHICH REMOTE CLIENT APPLICATION IS TO BE SHARED
411
NO
BUILD REMOTE WINDOW TREE STRUCTURES WITH REMOTE X SERVER PROCESS AND SYNCHRONIZE TREE STRUCTURE IN THE REMOTE SHAREDAPP PROCESS (FIG. 7B)
412
REMOTE SHAREDAPP STARTS SHARING EVENTS (FIG. 7D)
413
YES
CONTINUE SHARING APPLICATION?
414
NO

FIG. 7B

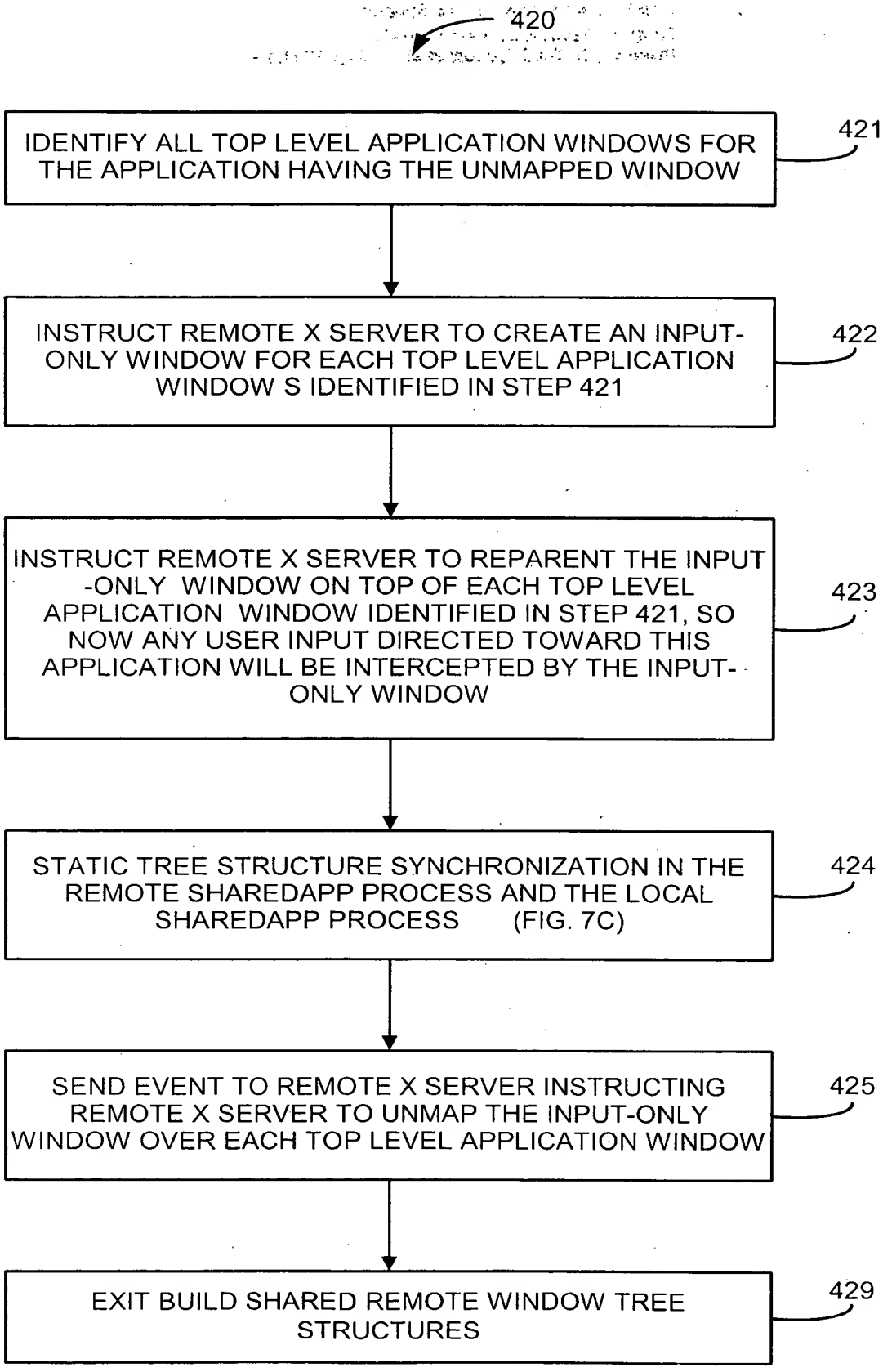

420
IDENTIFY ALL TOP LEVEL APPLICATION WINDOWS FOR THE APPLICATION HAVING THE UNMAPPED WINDOW
421
INSTRUCT REMOTE X SERVER TO CREATE AN INPUT-ONLY WINDOW FOR EACH TOP LEVEL APPLICATION WINDOW S IDENTIFIED IN STEP 421
422
INSTRUCT REMOTE X SERVER TO REPARENT THE INPUT -ONLY WINDOW ON TOP OF EACH TOP LEVEL APPLICATION WINDOW IDENTIFIED IN STEP 421, SO NOW ANY USER INPUT DIRECTED TOWARD THIS APPLICATION WILL BE INTERCEPTED BY THE INPUT-ONLY WINDOW
423
STATIC TREE STRUCTURE SYNCHRONIZATION IN THE REMOTE SHAREDAPP PROCESS AND THE LOCAL SHAREDAPP PROCESS (FIG. 7C)
424
SEND EVENT TO REMOTE X SERVER INSTRUCTING REMOTE X SERVER TO UNMAP THE INPUT-ONLY WINDOW OVER EACH TOP LEVEL APPLICATION WINDOW
425
EXIT BUILD SHARED REMOTE WINDOW TREE STRUCTURES
429

FIG. 7C

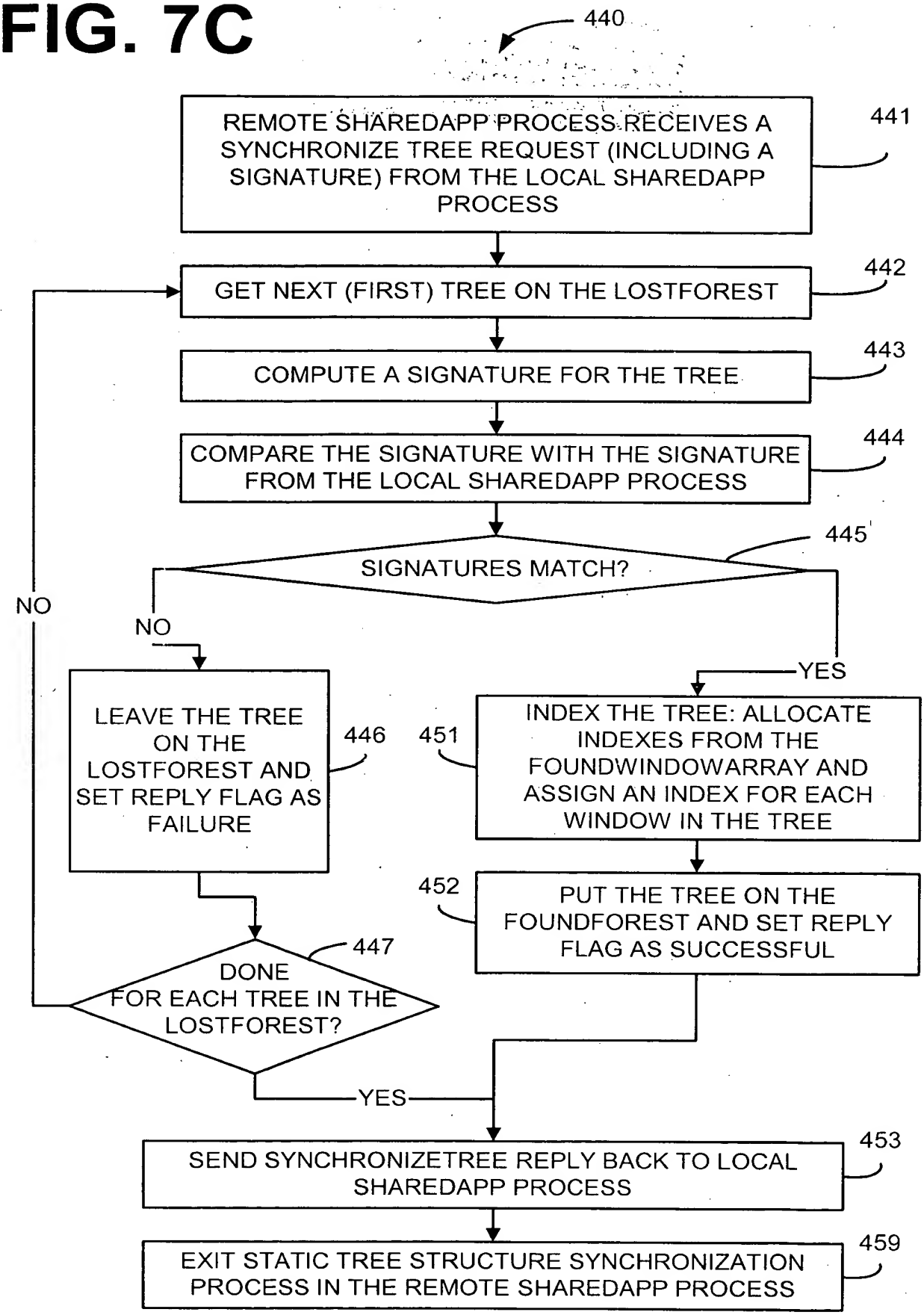

440
REMOTE SHAREDAPP PROCESS RECEIVES A SYNCHRONIZE TREE REQUEST (INCLUDING A SIGNATURE) FROM THE LOCAL SHAREDAPP PROCESS
441
GET NEXT (FIRST) TREE ON THE LOSTFOREST
442
COMPUTE A SIGNATURE FOR THE TREE
443
COMPARE THE SIGNATURE WITH THE SIGNATURE FROM THE LOCAL SHAREDAPP PROCESS
444
SIGNATURES MATCH?
445
NO
YES
NO
LEAVE THE TREE ON THE LOSTFOREST AND SET REPLY FLAG AS FAILURE
446
INDEX THE TREE: ALLOCATE INDEXES FROM THE FOUNDWINDOWARRAY AND ASSIGN AN INDEX FOR EACH WINDOW IN THE TREE
451
PUT THE TREE ON THE FOUNDFOREST AND SET REPLY FLAG AS SUCCESSFUL
452
DONE FOR EACH TREE IN THE LOSTFOREST?
447
YES
SEND SYNCHRONIZETREE REPLY BACK TO LOCAL SHAREDAPP PROCESS
453
EXIT STATIC TREE STRUCTURE SYNCHRONIZATION PROCESS IN THE REMOTE SHAREDAPP PROCESS
459

FIG. 7D

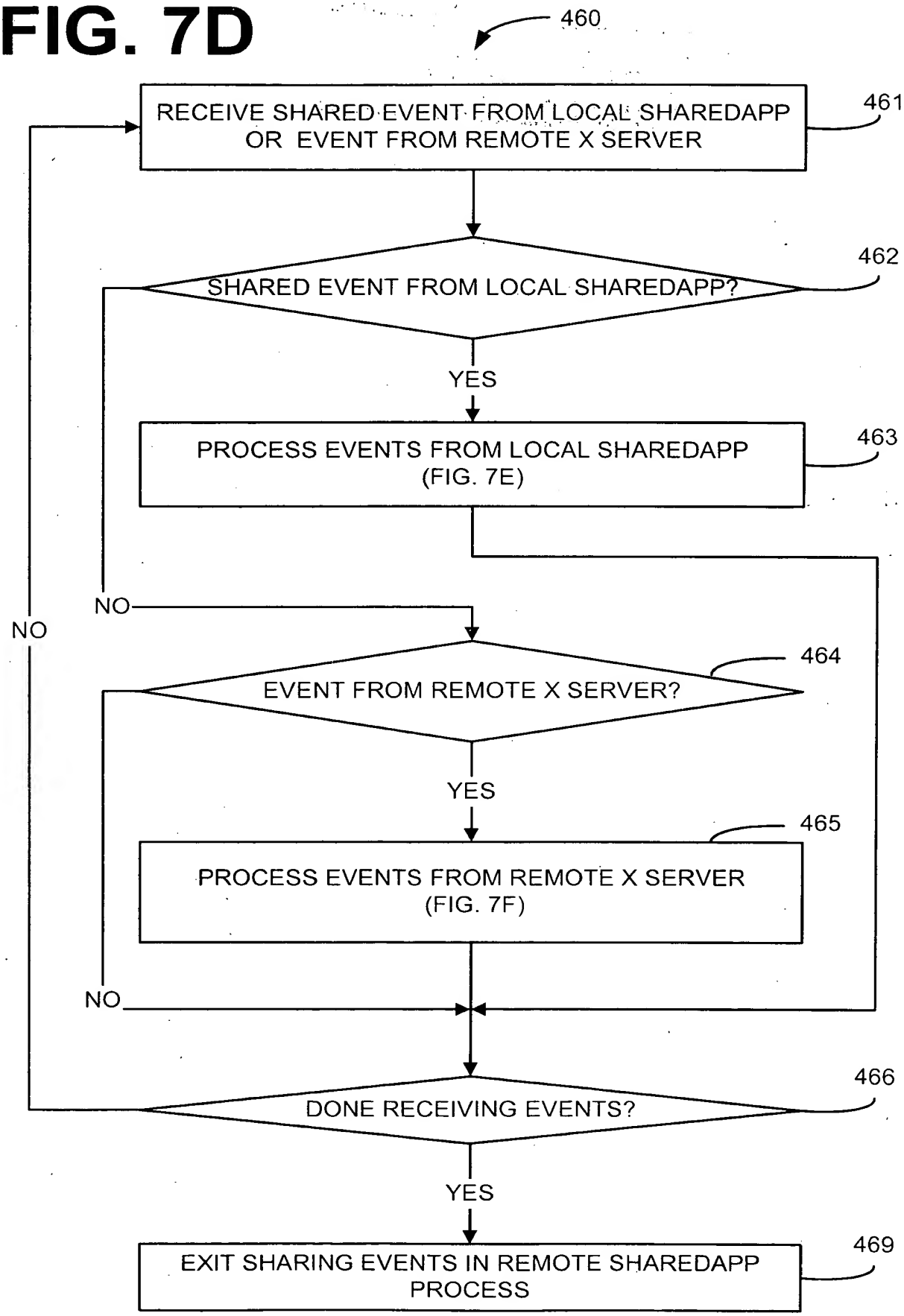

460
RECEIVE SHARED EVENT FROM LOCAL SHAREDAPP OR EVENT FROM REMOTE X SERVER
461
SHARED EVENT FROM LOCAL SHAREDAPP?
462
YES
PROCESS EVENTS FROM LOCAL SHAREDAPP (FIG. 7E)
463
NO
NO
EVENT FROM REMOTE X SERVER?
464
YES
PROCESS EVENTS FROM REMOTE X SERVER (FIG. 7F)
465
NO
DONE RECEIVING EVENTS?
466
YES
EXIT SHARING EVENTS IN REMOTE SHAREDAPP PROCESS
469

FIG. 7E

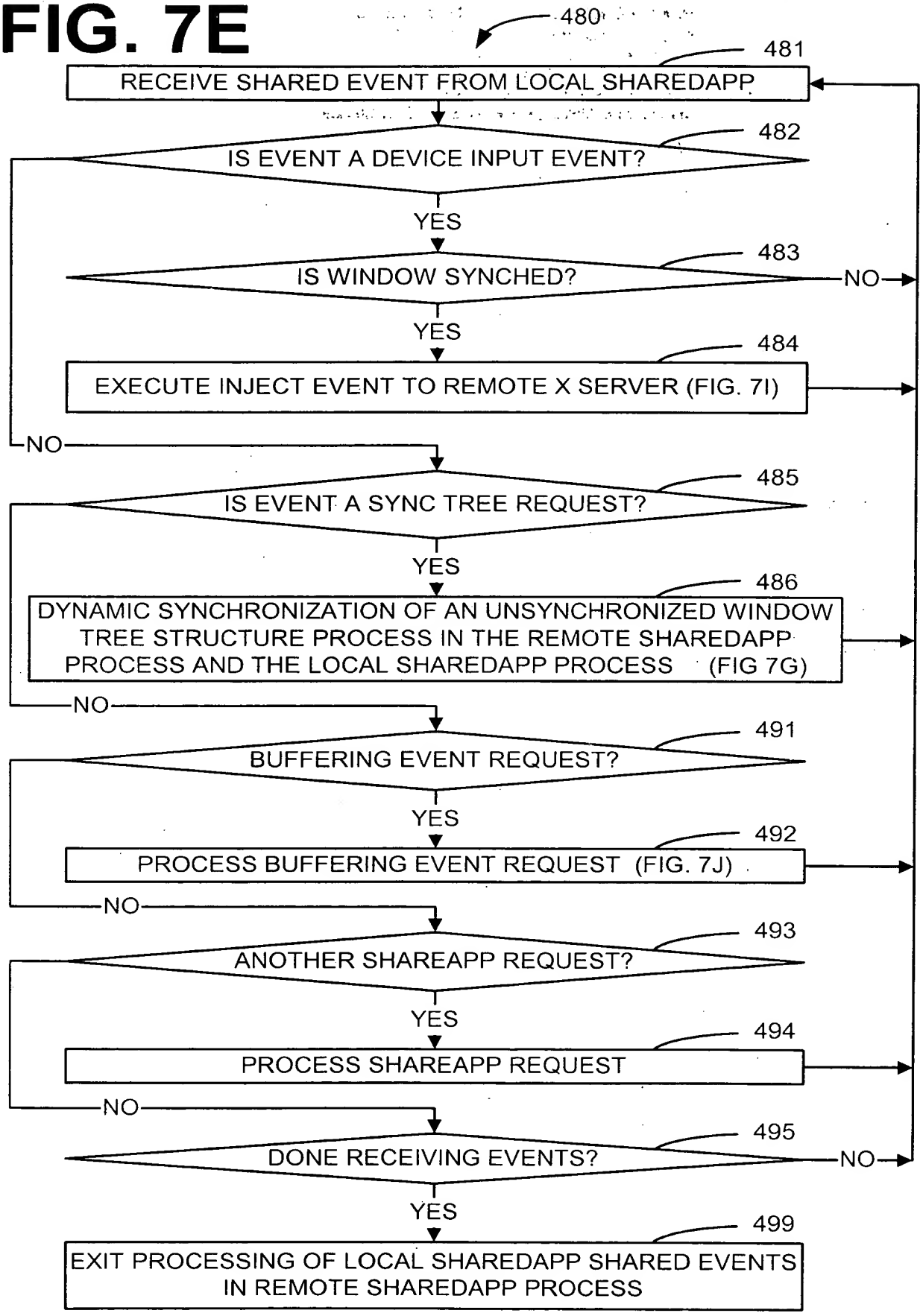

480
481
RECEIVE SHARED EVENT FROM LOCAL SHAREDAPP
482
IS EVENT A DEVICE INPUT EVENT?
YES
483
IS WINDOW SYNCHED?
NO
YES
484
EXECUTE INJECT EVENT TO REMOTE X SERVER (FIG. 7I)
NO
485
IS EVENT A SYNC TREE REQUEST?
YES
486
DYNAMIC SYNCHRONIZATION OF AN UNSYNCHRONIZED WINDOW TREE STRUCTURE PROCESS IN THE REMOTE SHAREDAPP PROCESS AND THE LOCAL SHAREDAPP PROCESS (FIG 7G)
NO
491
BUFFERING EVENT REQUEST?
YES
492
PROCESS BUFFERING EVENT REQUEST (FIG. 7J)
NO
493
ANOTHER SHAREAPP REQUEST?
YES
494
PROCESS SHAREAPP REQUEST
NO
495
DONE RECEIVING EVENTS?
NO
YES
499
EXIT PROCESSING OF LOCAL SHAREDAPP SHARED EVENTS IN REMOTE SHAREDAPP PROCESS

FIG. 7F

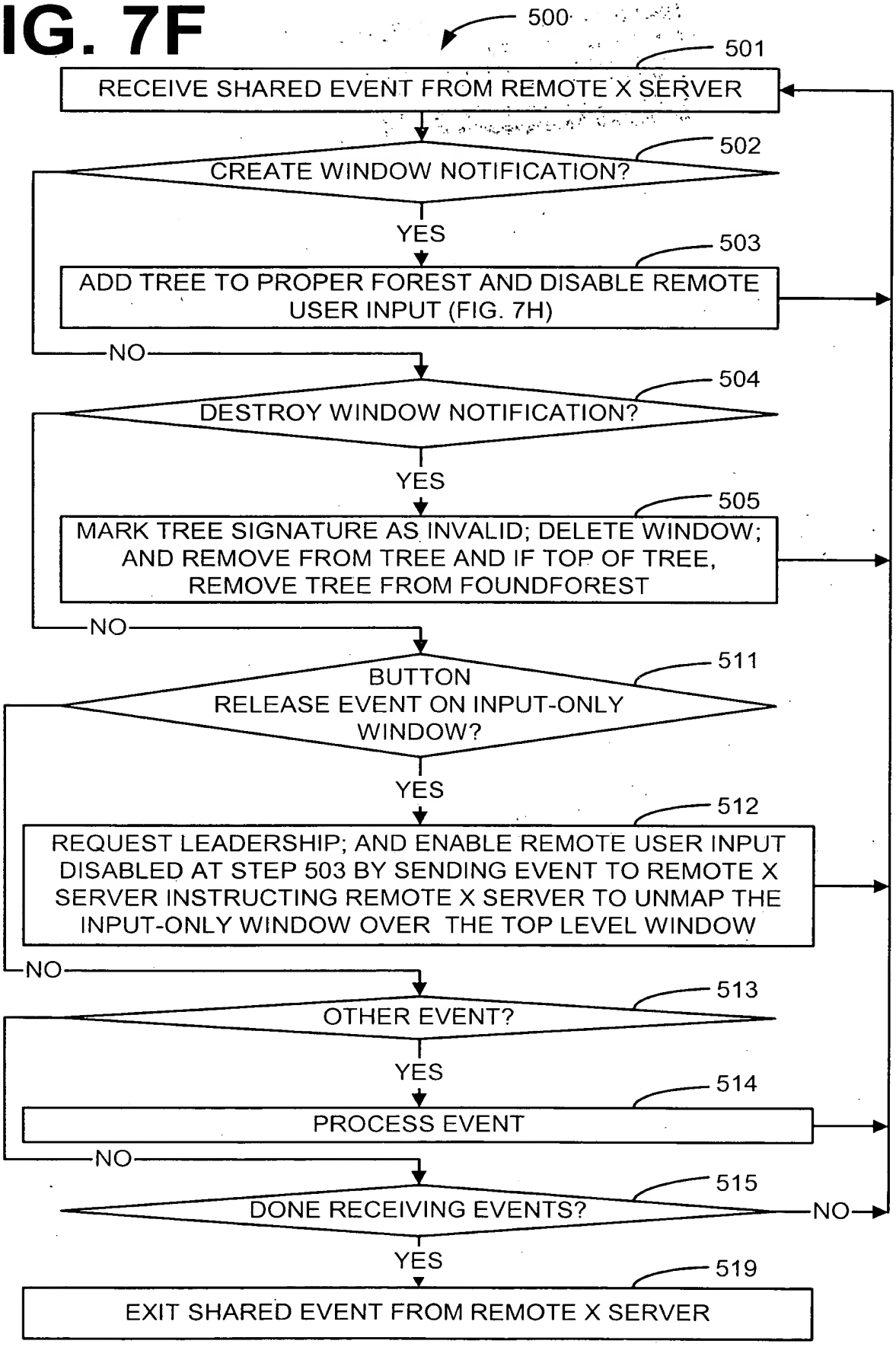

500
501
RECEIVE SHARED EVENT FROM REMOTE X SERVER
502
CREATE WINDOW NOTIFICATION?
YES
503
ADD TREE TO PROPER FOREST AND DISABLE REMOTE USER INPUT (FIG. 7H)
NO
504
DESTROY WINDOW NOTIFICATION?
YES
505
MARK TREE SIGNATURE AS INVALID; DELETE WINDOW; AND REMOVE FROM TREE AND IF TOP OF TREE, REMOVE TREE FROM FOUNDFOREST
NO
511
BUTTON RELEASE EVENT ON INPUT-ONLY WINDOW?
YES
512
REQUEST LEADERSHIP; AND ENABLE REMOTE USER INPUT DISABLED AT STEP 503 BY SENDING EVENT TO REMOTE X SERVER INSTRUCTING REMOTE X SERVER TO UNMAP THE INPUT-ONLY WINDOW OVER THE TOP LEVEL WINDOW
NO
513
OTHER EVENT?
YES
514
PROCESS EVENT
NO
515
DONE RECEIVING EVENTS?
NO
YES
519
EXIT SHARED EVENT FROM REMOTE X SERVER

FIG. 7G

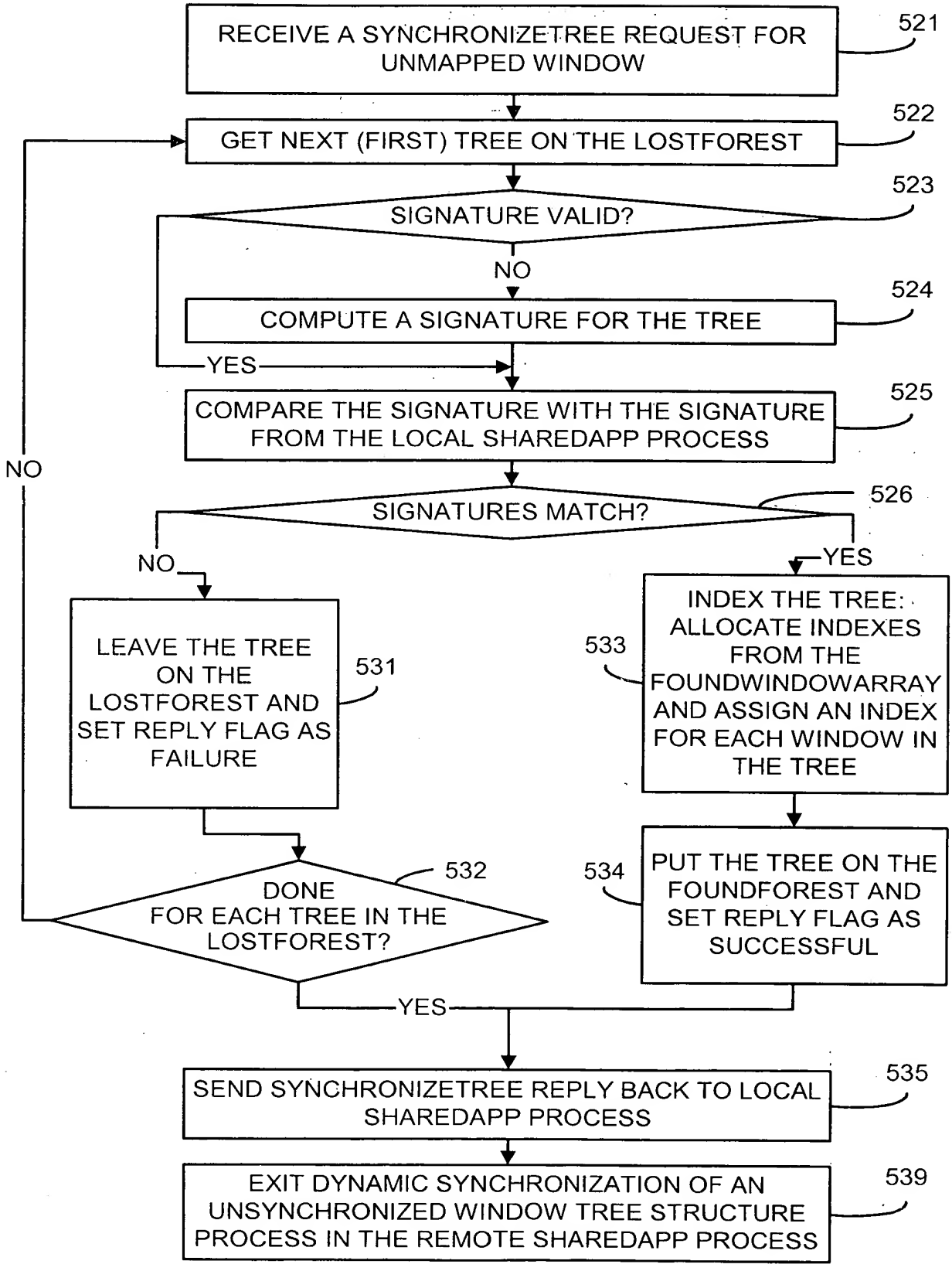

520
RECEIVE A SYNCHRONIZETREE REQUEST FOR UNMAPPED WINDOW
521
GET NEXT (FIRST) TREE ON THE LOSTFOREST
522
SIGNATURE VALID?
523
NO
COMPUTE A SIGNATURE FOR THE TREE
524
YES
COMPARE THE SIGNATURE WITH THE SIGNATURE FROM THE LOCAL SHAREDAPP PROCESS
525
SIGNATURES MATCH?
526
NO
YES
LEAVE THE TREE ON THE LOSTFOREST AND SET REPLY FLAG AS FAILURE
531
INDEX THE TREE: ALLOCATE INDEXES FROM THE FOUNDWINDOWARRAY AND ASSIGN AN INDEX FOR EACH WINDOW IN THE TREE
533
PUT THE TREE ON THE FOUNDFOREST AND SET REPLY FLAG AS SUCCESSFUL
534
DONE FOR EACH TREE IN THE LOSTFOREST?
532
NO
YES
SEND SYNCHRONIZETREE REPLY BACK TO LOCAL SHAREDAPP PROCESS
535
EXIT DYNAMIC SYNCHRONIZATION OF AN UNSYNCHRONIZED WINDOW TREE STRUCTURE PROCESS IN THE REMOTE SHAREDAPP PROCESS
539

FIG. 7H

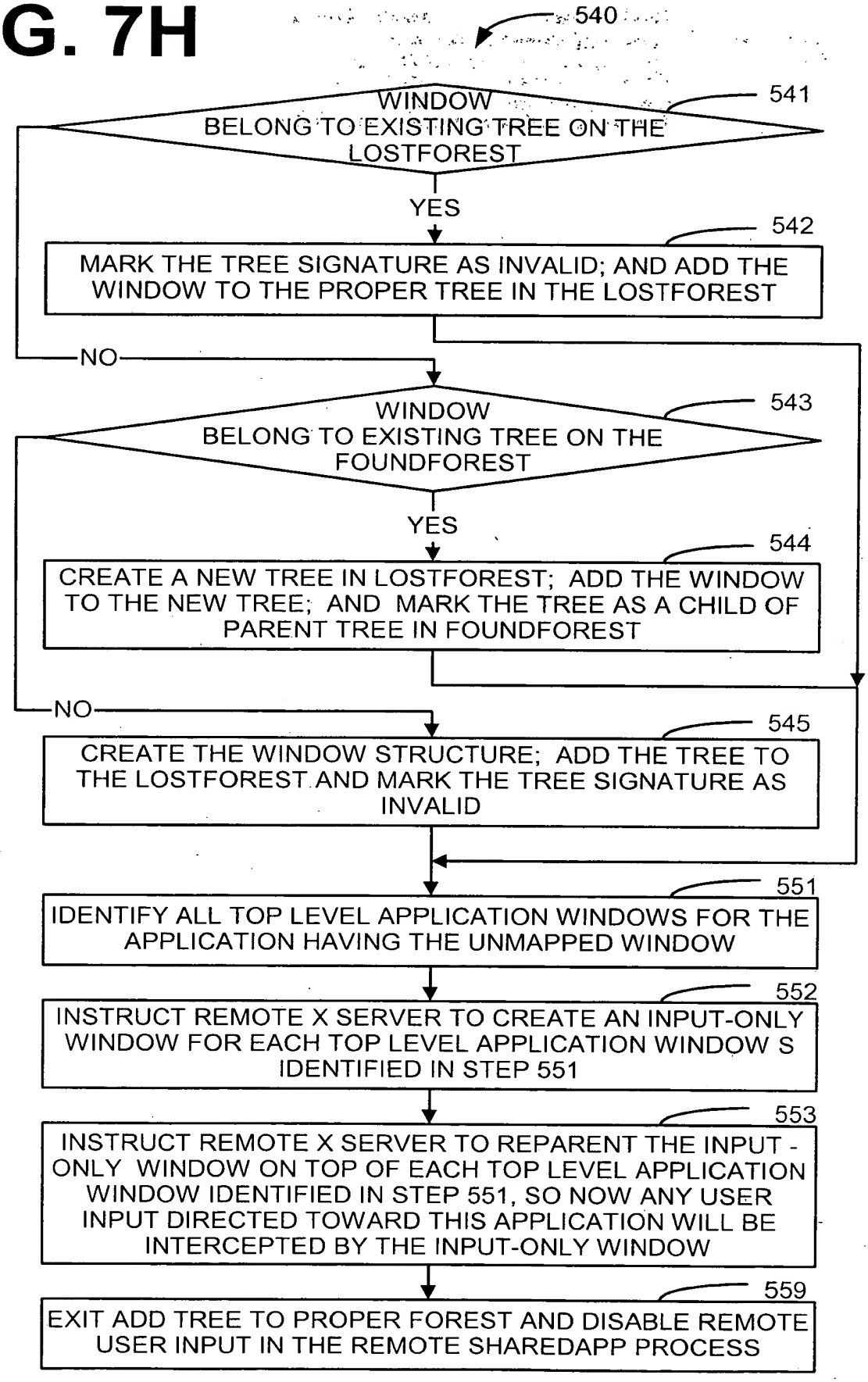

540
WINDOW BELONG TO EXISTING TREE ON THE LOSTFOREST
541
YES
MARK THE TREE SIGNATURE AS INVALID; AND ADD THE WINDOW TO THE PROPER TREE IN THE LOSTFOREST
542
NO
WINDOW BELONG TO EXISTING TREE ON THE FOUNDFOREST
543
YES
CREATE A NEW TREE IN LOSTFOREST; ADD THE WINDOW TO THE NEW TREE; AND MARK THE TREE AS A CHILD OF PARENT TREE IN FOUNDFOREST
544
NO
CREATE THE WINDOW STRUCTURE; ADD THE TREE TO THE LOSTFOREST AND MARK THE TREE SIGNATURE AS INVALID
545
IDENTIFY ALL TOP LEVEL APPLICATION WINDOWS FOR THE APPLICATION HAVING THE UNMAPPED WINDOW
551
INSTRUCT REMOTE X SERVER TO CREATE AN INPUT-ONLY WINDOW FOR EACH TOP LEVEL APPLICATION WINDOW S IDENTIFIED IN STEP 551
552
INSTRUCT REMOTE X SERVER TO REPARENT THE INPUT - ONLY WINDOW ON TOP OF EACH TOP LEVEL APPLICATION WINDOW IDENTIFIED IN STEP 551, SO NOW ANY USER INPUT DIRECTED TOWARD THIS APPLICATION WILL BE INTERCEPTED BY THE INPUT-ONLY WINDOW
553
EXIT ADD TREE TO PROPER FOREST AND DISABLE REMOTE USER INPUT IN THE REMOTE SHAREDAPP PROCESS
559

FIG. 7I

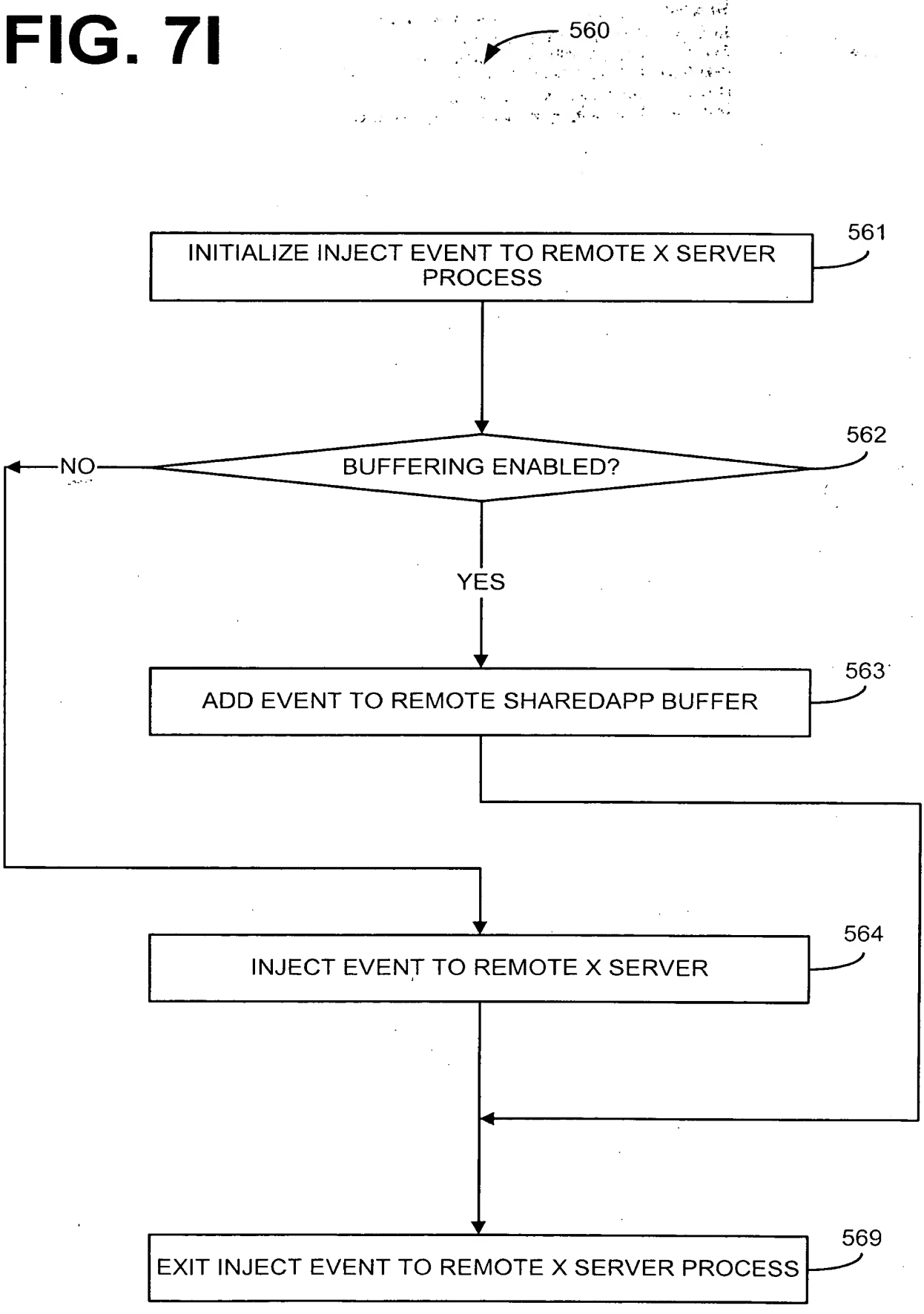

560
INITIALIZE INJECT EVENT TO REMOTE X SERVER PROCESS
561
BUFFERING ENABLED?
562
NO
YES
ADD EVENT TO REMOTE SHAREDAPP BUFFER
563
INJECT EVENT TO REMOTE X SERVER
564
EXIT INJECT EVENT TO REMOTE X SERVER PROCESS
569

FIG. 7J

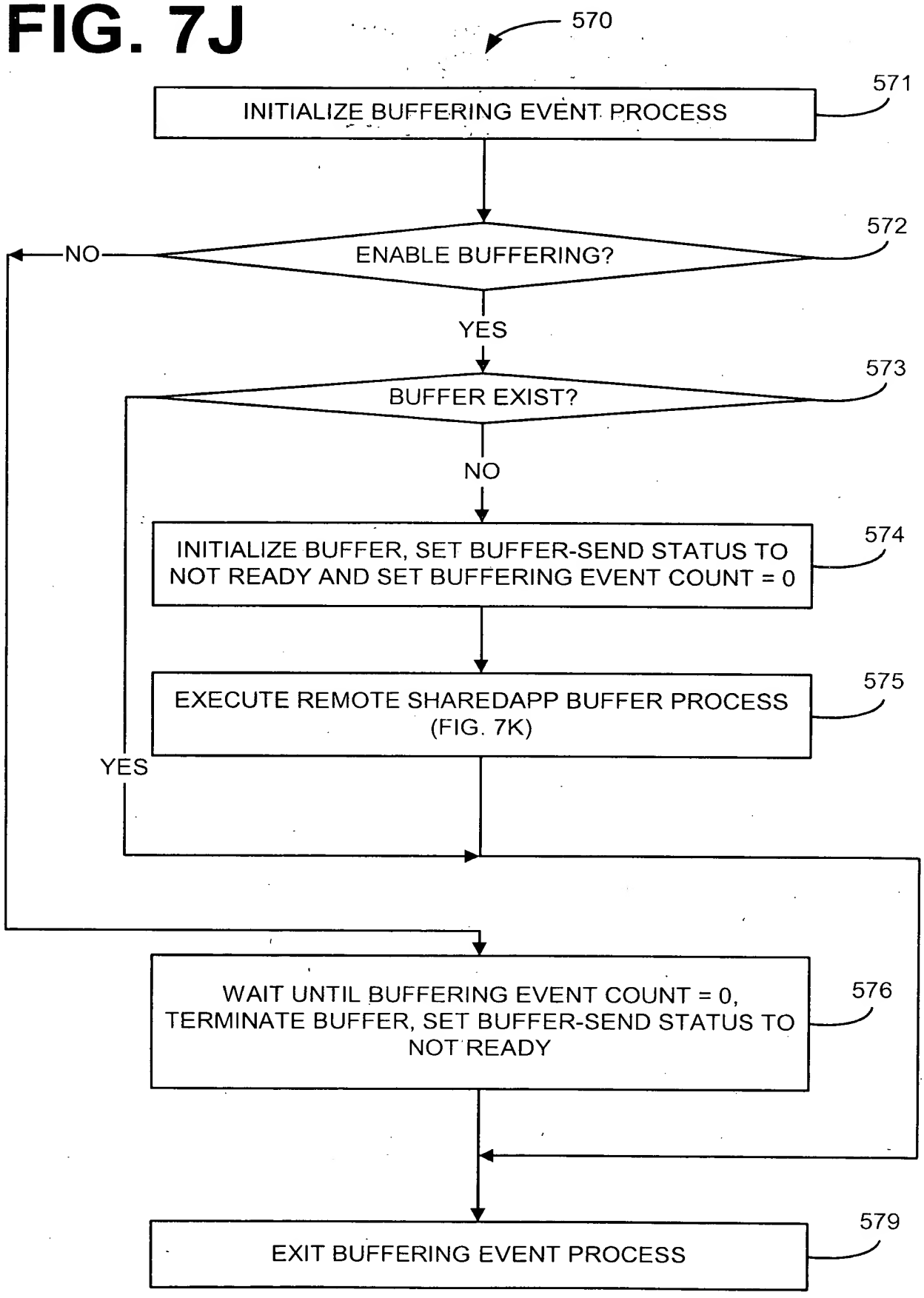

570
INITIALIZE BUFFERING EVENT PROCESS
571
ENABLE BUFFERING?
572
NO
YES
BUFFER EXIST?
573
NO
INITIALIZE BUFFER, SET BUFFER-SEND STATUS TO NOT READY AND SET BUFFERING EVENT COUNT = 0
574
EXECUTE REMOTE SHAREDAPP BUFFER PROCESS (FIG. 7K)
575
YES
WAIT UNTIL BUFFERING EVENT COUNT = 0, TERMINATE BUFFER, SET BUFFER-SEND STATUS TO NOT READY
576
EXIT BUFFERING EVENT PROCESS
579

FIG. 7K

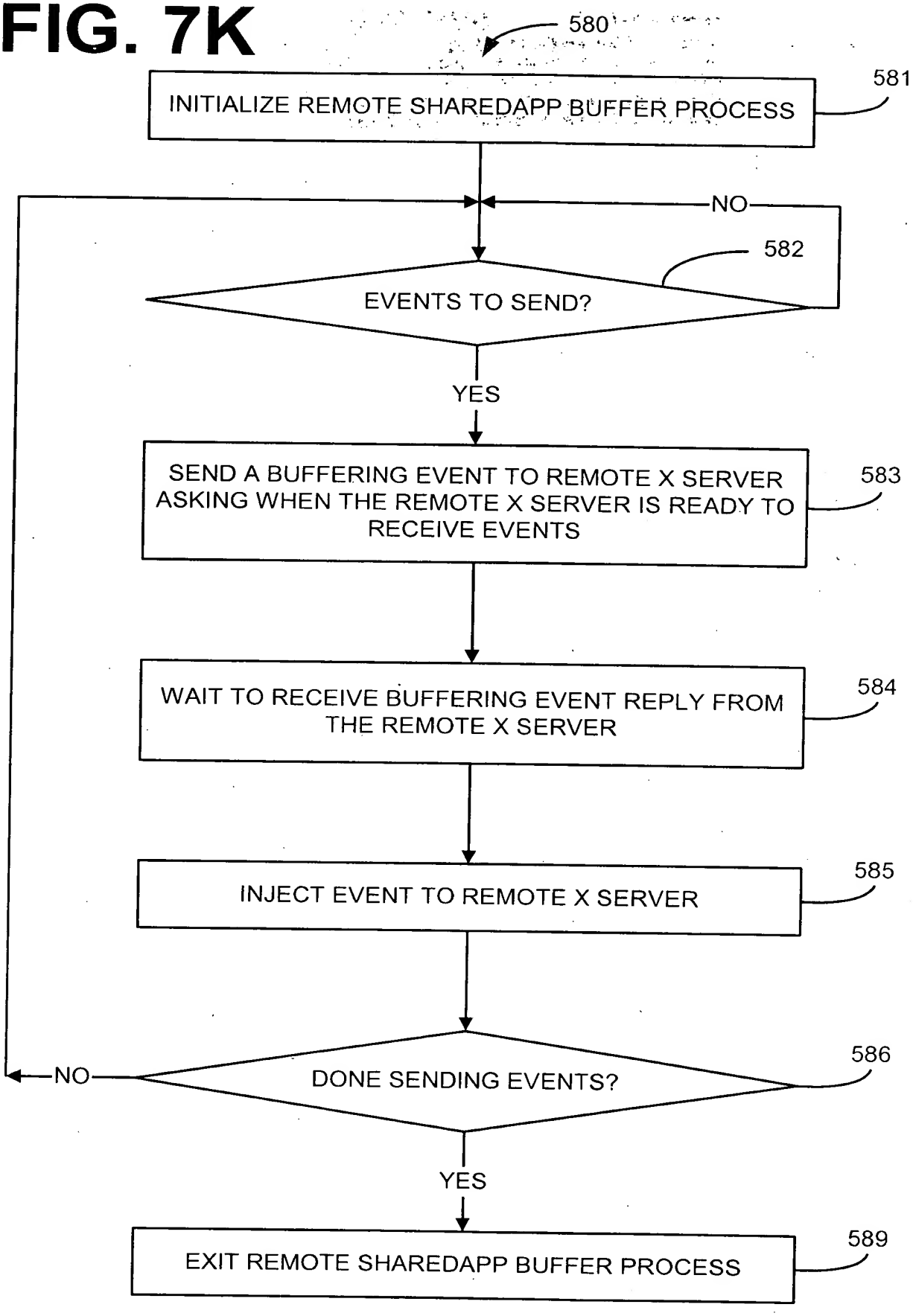

580
INITIALIZE REMOTE SHAREDAPP BUFFER PROCESS
581
NO
582
EVENTS TO SEND?
YES
SEND A BUFFERING EVENT TO REMOTE X SERVER ASKING WHEN THE REMOTE X SERVER IS READY TO RECEIVE EVENTS
583
WAIT TO RECEIVE BUFFERING EVENT REPLY FROM THE REMOTE X SERVER
584
INJECT EVENT TO REMOTE X SERVER
585
NO
DONE SENDING EVENTS?
586
YES
EXIT REMOTE SHAREDAPP BUFFER PROCESS
589

FIG. 8A

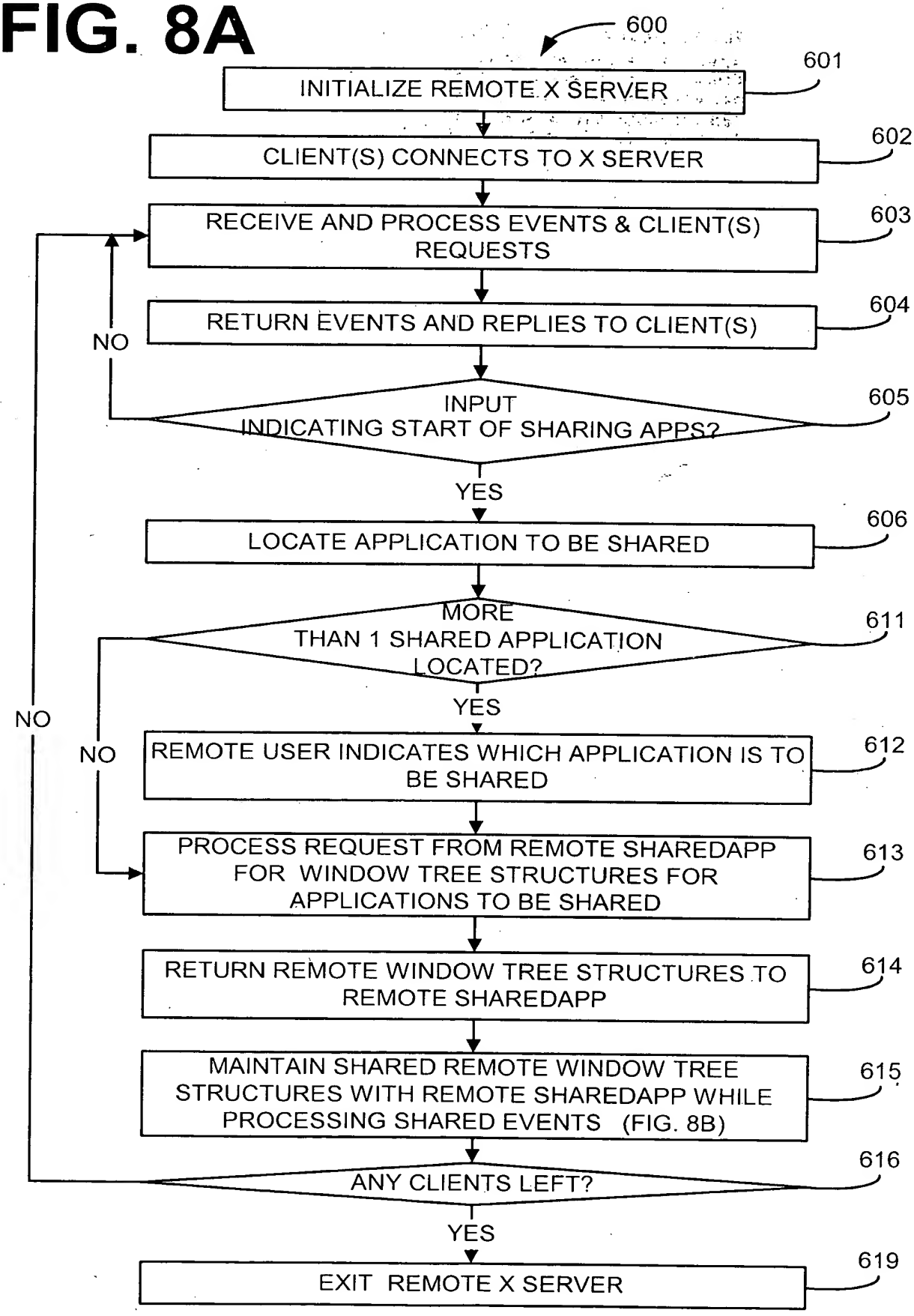
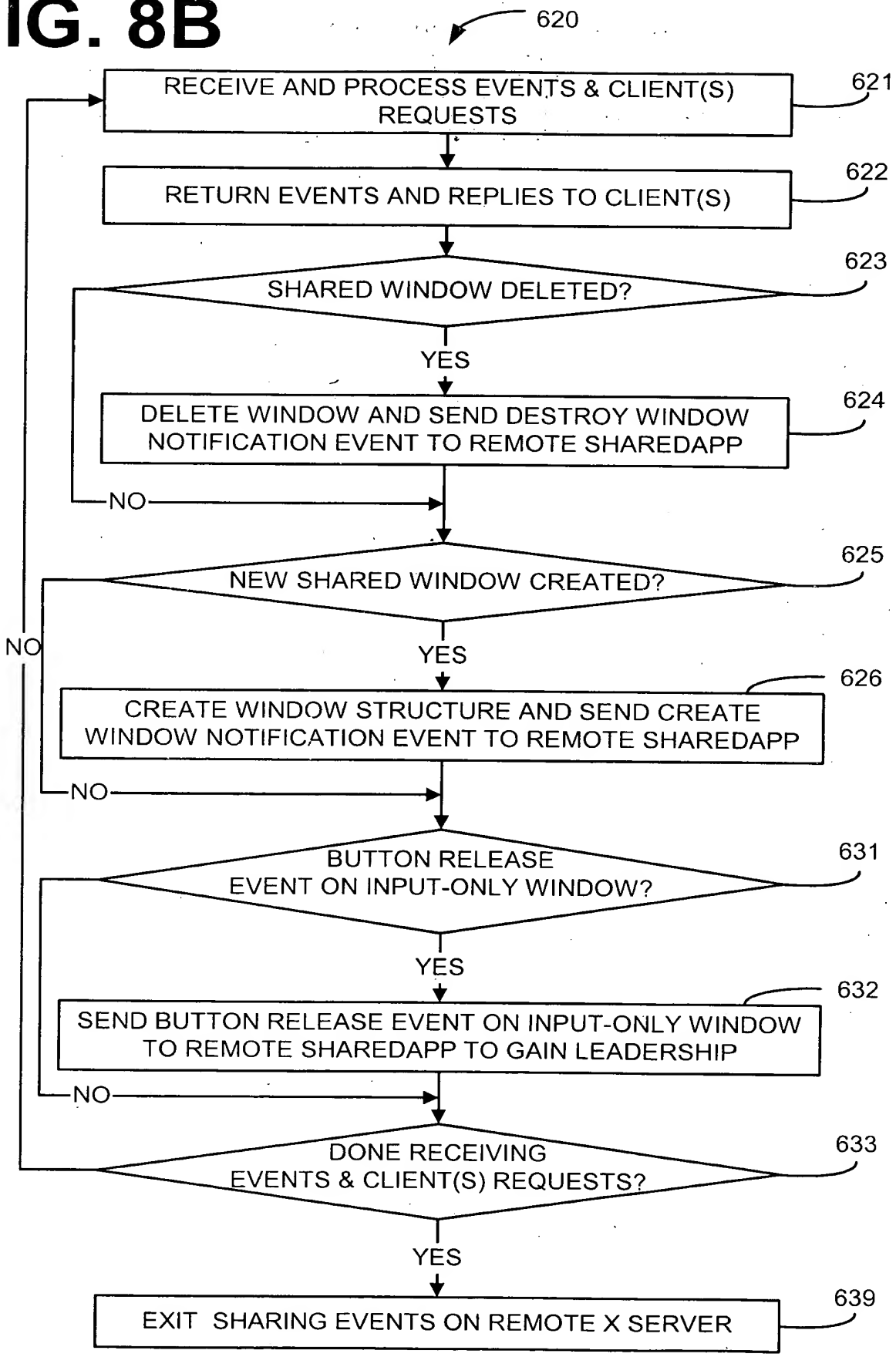

600
INITIALIZE REMOTE X SERVER
601
CLIENT(S) CONNECTS TO X SERVER
602
RECEIVE AND PROCESS EVENTS & CLIENT(S) REQUESTS
603
RETURN EVENTS AND REPLIES TO CLIENT(S)
604
INPUT INDICATING START OF SHARING APPS?
605
NO
YES
LOCATE APPLICATION TO BE SHARED
606
MORE THAN 1 SHARED APPLICATION LOCATED?
611
NO
YES
REMOTE USER INDICATES WHICH APPLICATION IS TO BE SHARED
612
NO
PROCESS REQUEST FROM REMOTE SHAREDAPP FOR WINDOW TREE STRUCTURES FOR APPLICATIONS TO BE SHARED
613
RETURN REMOTE WINDOW TREE STRUCTURES TO REMOTE SHAREDAPP
614
MAINTAIN SHARED REMOTE WINDOW TREE STRUCTURES WITH REMOTE SHAREDAPP WHILE PROCESSING SHARED EVENTS (FIG. 8B)
615
ANY CLIENTS LEFT?
616
YES
EXIT REMOTE X SERVER
619

FIG. 8B


620
RECEIVE AND PROCESS EVENTS & CLIENT(S) REQUESTS
621
RETURN EVENTS AND REPLIES TO CLIENT(S)
622
SHARED WINDOW DELETED?
623
YES
DELETE WINDOW AND SEND DESTROY WINDOW NOTIFICATION EVENT TO REMOTE SHAREDAPP
624
NO
NEW SHARED WINDOW CREATED?
625
NO
YES
CREATE WINDOW STRUCTURE AND SEND CREATE WINDOW NOTIFICATION EVENT TO REMOTE SHAREDAPP
626
NO
BUTTON RELEASE EVENT ON INPUT-ONLY WINDOW?
631
YES
SEND BUTTON RELEASE EVENT ON INPUT-ONLY WINDOW TO REMOTE SHAREDAPP TO GAIN LEADERSHIP
632
NO
DONE RECEIVING EVENTS & CLIENT(S) REQUESTS?
633
YES
EXIT SHARING EVENTS ON REMOTE X SERVER
639